Multibody Dynamics 2015

Barcelona, June 29 – July 2
Barcelona School of Industrial Engineering
Universitat Politècnica de Catalunya

multibody2015.org
# TABLE OF CONTENTS

Preface .................................................................................................................. 7
Supporting Organizations and Industrial Sponsors................................. 9
Contents .............................................................................................................. 11
Formulations and Numerical Methods ...................................................... 23
Efficient Methods and Real-Time Applications................................. 164
Flexible Multibody Dynamics ................................................................. 241
Contact Dynamics and Constraints ......................................................... 491
Multiphysics and Coupled Problems .................................................... 609
Control and Optimization ........................................................................ 720
Software Development and Computer Technology .......................... 860
Aerospace and Maritime Applications .................................................. 895
Biomechanics .............................................................................................. 978
Railroad Vehicle Dynamics ................................................................. 1101
Road Vehicle Dynamics .......................................................................... 1221
Robotics ........................................................................................................ 1388
Benchmark Problems .............................................................................. 1538
Authors Index .............................................................................................. 1631
This volume contains the full papers accepted for presentation at the ECCOMAS Thematic Conference on Multibody Dynamics 2015 held in the Barcelona School of Industrial Engineering, Universitat Politècnica de Catalunya, on June 29 - July 2, 2015.

The ECCOMAS Thematic Conference on Multibody Dynamics is an international meeting held once every two years in a European country. Continuing the very successful series of past conferences that have been organized in Lisbon (2003), Madrid (2005), Milan (2007), Warsaw (2009), Brussels (2011) and Zagreb (2013); this edition will once again serve as a meeting point for the international researchers, scientists and experts from academia, research laboratories and industry working in the area of multibody dynamics.

Applications are related to many fields of contemporary engineering, such as vehicle and railway systems, aeronautical and space vehicles, robotic manipulators, mechatronic and autonomous systems, smart structures, biomechanical systems and nanotechnologies. The topics of the conference include, but are not restricted to:

- Formulations and Numerical Methods
- Efficient Methods and Real-Time Applications
- Flexible Multibody Dynamics
- Contact Dynamics and Constraints
- Multiphysics and Coupled Problems
- Control and Optimization
- Software Development and Computer Technology
- Aerospace and Maritime Applications
- Biomechanics
- Railroad Vehicle Dynamics
- Road Vehicle Dynamics
- Robotics
- Benchmark Problems

The conference is organized by the Department of Mechanical Engineering of the Universitat Politècnica de Catalunya (UPC) in Barcelona. The organizers would like to thank the authors for submitting their contributions, the keynote lecturers for accepting the invitation and for the quality of their talks, the awards and scientific committees for their support to the organization of the conference, and finally the topic organizers for reviewing all extended abstracts and selecting the awards nominees.

Josep M. Font-Llagunes
Department of Mechanical Engineering
Barcelona School of Industrial Engineering
Universitat Politècnica de Catalunya
Barcelona, Catalonia, Spain
SUPPORTING ORGANIZATIONS AND INDUSTRIAL SPONSORS

Supporting Organizations

[Logos of ECCOMAS, IFToMM, IUTAM, JSME, KSME, CIMNE]

Industrial Sponsors

[Logos of Maplesoft, RECURDYN, HP, FUNDACIÓN CIM]
<table>
<thead>
<tr>
<th>Title</th>
<th>Authors</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Formulations and Numerical Methods</td>
<td></td>
<td></td>
</tr>
<tr>
<td>A weak formulation for the numerical solution of the equations of motion of multibody systems subjected to scleronomic constraints</td>
<td>N. Potosakis, E. Paraskevopoulos and S. Natsiavas</td>
<td>23</td>
</tr>
<tr>
<td>Adaptive meshing in two-dimensional beam elements based on the absolute nodal coordinate formulation</td>
<td>A.I. Valkeapää and A.M. Mikkola</td>
<td>35</td>
</tr>
<tr>
<td>An approach to the parameterization of dynamic models of serial mechanisms</td>
<td>S. Krutikov</td>
<td>43</td>
</tr>
<tr>
<td>Capturing complex nonlinear failure of bolt connections with simple multibody system models</td>
<td>D. Neuenhaus, M. Feldmann and U. Geißler</td>
<td>55</td>
</tr>
<tr>
<td>Conceptual and numerical aspects of the mixed variational formulation of geometrically exact beam models</td>
<td>P. Češarek, D. Zupan and J. Linn</td>
<td>66</td>
</tr>
<tr>
<td>Geometrically based pseudo-inverse and reciprocal screws of the Jacobian of 4H mechanisms</td>
<td>R. Bartkowiak and C. Woernle</td>
<td>77</td>
</tr>
<tr>
<td>Multibody kinematics. A topological formulation based on structural-group coordinates</td>
<td>M. Saura, P. Segado, B. Muñoz, D. Dopico and J. Cuadrado</td>
<td>88</td>
</tr>
<tr>
<td>Nonlinear random vibration of the cable modeled as MDOF system and excited by filtered Gaussian white noise</td>
<td>G. K. Er, V. P. Iu and K. Wang</td>
<td>100</td>
</tr>
<tr>
<td>Numerical integration of underactuated mechanical systems subjected to mixed holonomic and servo constraints</td>
<td>P. Betsch, R. Altmann and Y. Yang</td>
<td>109</td>
</tr>
<tr>
<td>Relationship between the natural and relative coordinates through the DeNOC matrices</td>
<td>A.K. Sharma and S.K. Saha</td>
<td>122</td>
</tr>
<tr>
<td>The accuracy of integrated rotational quaternion and angular velocity from curvature</td>
<td>A. Treven and M. Saje</td>
<td>132</td>
</tr>
<tr>
<td>The logarithmic complexity procedure for parallel multibody dynamics solution</td>
<td>L. Mraz and M. Valasek</td>
<td>143</td>
</tr>
<tr>
<td>Various multibody dynamic models for the description of plane Kirchhoff rods</td>
<td>H. Lang, H. Laube and S. Leyendecker</td>
<td>152</td>
</tr>
</tbody>
</table>
Efficient Methods and Real-Time Applications

Development, implementation and validation of a hard real-time multibody simulation for high-fidelity steering wheel force feedback
R. Pastorino, M. Desloovere, F. Vanneste, P. Degezelle and W. Desmet ........................................ 164

Enhancing the performance of the divide-and-conquer algorithm when forming and solving the equations of motion for multibody systems
J.J. Laflin, K.S. Anderson and M. Hans ................................................................................................. 176

Forward dynamics of variable topology mechanisms - The case of constraint activation
A. Müllner ................................................................................................................................................ 184

Generalization of the divide-and-conquer algorithm for the uncertainty quantification of multibody systems
M. Poursina ............................................................................................................................................. 196

Multibody model reduction by parameter elimination
X. Iriarte, J. Ros, V. Mata and A. Plaza .................................................................................................. 208

Parallel recursive Hamiltonian formulation for constrained multibody system dynamics
K. Chadaj, P. Malczyk and J. Frączek .................................................................................................... 217

Simultaneous estimation of kinematic state and unknown input forces in rigid-link multibody systems
I. Palomba, D. Richiedei and A. Trevisani ............................................................................................ 229

Flexible Multibody Dynamics

An analysis method for a system with mass and extremely flexible component and its application to analysis of deployable satellite
Y. Sugawara, S. Oshima and T. Chida .................................................................................................. 241

Approximative modeling of compliant mechanism dynamics
M. Rösner, R. Lammering and R. Friedrich .......................................................................................... 253

Compatibility equation and revision on the expression of chord-line component of dynamical cable tension
G.K. Er, K. Wang and V.P. Iu .................................................................................................................. 265

Deployment dynamics of a large-scale ring truss mesh antennas
P. Li, C. Liu, Q. Tian and H. Hu ............................................................................................................ 274

Development of efficient flexible multibody techniques for rotordynamic systems including rotors and supporting structures
S. Rossin, E. Meli, G. Pallini, S. Panconi and A. Rindi ......................................................................... 286

Dynamic analysis of the rotating thin-walled structure with absolute nodal coordinate formulation
C. Zhao, H. Yu, H. Wang and Y. Zhao ................................................................................................. 297

Dynamic fracture simulation of flexible multibody systems via absolute nodal coordinate formulation and SPH method
W. Hu, Q. Tian and H. Hu ...................................................................................................................... 308
Dynamic modeling approach for a continuous moving belt
N. Lorenz, G. Offner, M. Sopouch and O. Ertl ................................................................. 320

Efficient formation and solution of equations of motion for multibody systems with bodies of mixed definition using DCA
J.J. Laflin, K.S. Anderson and I.M. Khan ........................................................................... 334

Eliminating interface degrees-of-freedom in substructuring
O. Kazemi and P.E. Nikravesh ............................................................................................. 347

Influence of the fibre spring-damper model in a simple laboratory mechanical system on the coincidence with the experimental results
P. Polach and M. Hajžman ..................................................................................................... 356

Interpolation and truncation model reduction techniques in coupled elastic multibody systems
P. Holzwarth and P. Eberhard ............................................................................................... 366

Interpolation-based parametric model order reduction for elastic multibody systems
M. Fischer and P. Eberhard .................................................................................................. 379

Investigation on the modeling of membranes in large scale multi-body systems using co-rotational formulation
J. Shi, Z. Liu and J. Hong ...................................................................................................... 392

New formulation of the rigid finite element method and its application to modelling of lines and risers
I. Adamiec-Wójcik, Ł. Drąg and S. Wojciech ...................................................................... 400

Nonlinear vibration of cantilever under the action of lateral harmonic excitations and axial load
G.K. Er and H.E. Du ............................................................................................................. 410

Quadratic manifolds for reduced order modelling of highly flexible multibody systems
L. Wu, P. Tiso, S. Jain and F. van Keulen .......................................................................... 422

Relative and absolute finite element nodal coordinates in the dynamics of large flexible structures subject to seismic excitations
E.V. Zahariev, V.S. Jivkov and I.D. Danchev ..................................................................... 431

Three-dimensional nonlinear shell theory for flexible multibody dynamics
S.L. Han and O.A. Bauchau ............................................................................................... 443

Topology optimization of bearing domains in flexible multibody systems
A. Moghadasi, A. Held and R. Seifried .............................................................................. 468

Towards viscoplastic constitutive models for Cosserat rods
V. Dörlich, J. Linn, T. Scheffer and S. Diebels .................................................................. 479
Contact Dynamics and Constraints

A mixed shooting and harmonic balance method for mechanical systems with dry friction or other local nonlinearities
F. Schreyer and R.I. Leine ................................................................. 491

A proposition of analysis method for the system with contact between largely deformable body and rigid body
T. Chida and Y. Sugawara ................................................................. 502

Contact problems using quaternions in Lagrange equations
R. Souchet ...................................................................................... 510

Damping amplification caused by a mechanism that trans-pass through its singular position
K. Lipinski ...................................................................................... 520

Dynamics of synchronization of rotational motion of contacting triple-body systems
V. Berbyuk ...................................................................................... 532

Forward dynamics of selected spatial one-dof linkage mechanism with Dahl friction model in revolute joints
A. Harlecki and A. Urbaś ........................................................................ 542

Nonlinear analysis of the cable-pulley interaction
M. Hajžman, R. Bulín and P. Polach ............................................................ 554

On the frictional contacts in multibody system dynamics
F. Marques, P. Flores and H.M. Lankarani .................................................. 565

Spatial impact of a beam attached to a sliding structure
M. Dupac ......................................................................................... 577

Synchronization-based state observer for impacting multibody systems using switched geometric unilateral constraints
M. Baumann and R.I. Leine .................................................................... 585

Towards a maximal monotone impact law for Newton’s cradle
T. Winandy and R.I. Leine .................................................................... 597

Multiphysics and Coupled Problems

A numerical study of an autoparametric system with added electromagnetic energy harvester
A. Mitura, K. Kecik, J. Warminski, W. Jarzyna and S. Lenci ................................. 609

A parametric approach to the generation of multidisperse granular flows for particle simulations
A. Tasora, I. Critelli and H. Mazhar .......................................................... 616

Coupled analysis of thermo-fluid-flexible multibody dynamics for a two-dimensional engine nozzle

Dynamic modeling of a 3D printer based on a four arms SCARA mechanism
E.P. Okabe and P. Masarati ...................................................................... 641
Index-2 co-simulation approach for solver coupling with algebraic constraints  
D. Lu, P. Li and B. Schweizer ........................................................................................................... 654

Numerical model for simulating cam shifting systems in automotive engines  
M. Busch, S. Wolf and C. Träbing ................................................................................................... 666

On the stability of explicit and implicit co-simulation approaches  
P. Li, D. Lu and B. Schweizer ........................................................................................................... 679

Simulations of a small wind turbine with a bolted lattice tower under consideration of mass distribution and damping  
R. Rachholz, C. Woernle, H. Panwitt and R. Staschko ..................................................................... 691

Variable communication steps and stiffness computation in co-simulation of large MBD and FEA systems  
I. Ishikawa, M. Tateishi and J.L. Ortiz .............................................................................................. 701

Virtual sensing on mechatronic drivetrains using multiphysical models  
B. Forrier, F. Naets and W. Desmet .................................................................................................. 708

Control and Optimization

A Kalman filter-based algorithm for IMU signals fusion applied to track geometry estimation  
A.F. Hidalgo, J. Ros and J.L. Escalona ............................................................................................... 720

A comparison between finite element models and MBS models in automotive safety applications  
J. Fehr and C. Kleinbach .................................................................................................................. 733

A simulation concept for the azimuth system of horizontal axis wind turbines  
J. Luthe, C. Woernle and J. Zierath ................................................................................................. 742

A transcription method for optimal control problems in multibody dynamics  
C. Becker and P. Betsch ................................................................................................................... 754

Application of optimisation and an artificial neural network to stabilisation of a load relocated under water  
I. Adamiec-Wójcik and Ł. Drąg .......................................................................................................... 766

Capabilities of model predictive control for load reduction on wind turbines  
J. Zierath, U. Jassmann and D. Abel .................................................................................................. 776

Energy optimal manipulation of an industrial robot  
T. Lauss, P. Leitner, S. Oberpeilsteiner and W. Steiner .................................................................... 789

Mechatronic solution of components cooperation in the device for gait reeducation  
S. Duda, G. Gembalczyk, S. Kciuk, J. Jurkojc and R. Michnik .......................................................... 799

Nonlinear dynamics and bifurcation of a vibration absorber-harvester system  
K. Kecik, P. Brzeski and P. Perlikowski ............................................................................................ 806
Optimal static balancing of a spatial palletizing robot  
R. Carrabotta, A. Martini, M. Troncossi and A. Rivola .......................................................... 817

Structural topology optimization of multibody systems  
T. Ghandriz, C. Führer and H. Elmqvist ................................................................. 828

Structure preserving optimal control of a 3d-dimensional upright gait  
M.W. Koch and S. Leyendecker ........................................................................ 839

Trajectory tracking problem for omnidirectional mobile robot with parameter variations and delayed feedback  
A.S. Andreev and O.A. Peregudova ........................................................................ 851

### Software Development and Computer Technology

Interests and complementary of two structural modeling tools for the analysis of multibody systems: Maplesim and bond graph with 20-sim  
B. Boudon and F. Malburet ................................................................................. 860

Issues in modelling a Stewart platform in multibody dynamics software  
V. Gupta, M.H. Koul and S.K. Saha .................................................................. 872

Robotran-Yarp interface: a framework for real-time controller development based on multibody dynamics simulation  

Verification of a numerical simulation code for underwater catenary chain  
X.Q. Zhu and W.S. Yoo .................................................................................. 886

### Aerospace and Maritime Applications

Analytical and numerical multibody dynamic modeling and constraint force analysis in an axial piston pump  
T. R. Milind and M. Mitra ................................................................................ 895

Development of 3D multibody models of Intervention-Autonomous Underwater Vehicle (I-AUV)  
B. Allotta, R. Conti, R. Costanzi, F. Fanelli, E. Meli and A. Ridolfi ............................. 904

Dynamics of multimass rotor with active hybrid bearings  
R.N. Polyakov, S. V. Mayorov, M. E. Bondarenko and L. A. Savin ............................ 912

Heterogeneous, multi-tier wheel ground contact for planetary exploration  
R. Lichtenheldt, M. Hellerer, S. Barthelmes and F. Buse ........................................ 924

Numerical simulation of n-body asteroid aggregation  
F. Ferrari, A. Tasora, P. Masarati and M. Lavagna ............................................. 936

Planning and control of autonomous underwater vehicle’s trajectory for its recovery from a mobile submarine platform  
P. Martinez, J.L. Aguirre, M. Saura, P. Segado and A. Luaces .................................... 948

Simulation of the flexible body moving in viscous fluid  
Y.A. Dynnikov, G.Y. Dynnikova and S.V. Guvernyuk ........................................ 960
<table>
<thead>
<tr>
<th>Title</th>
<th>Authors</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wave-based attitude control of spacecraft with fuel sloshing dynamics</td>
<td>J. W. Thompson and W. J. O'Connor</td>
<td>968</td>
</tr>
<tr>
<td>Biomechanics</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Comparison of control and optimization approaches for trajectory</td>
<td>Á. Noriega, U. Lugrís and J. Cuadrado</td>
<td>978</td>
</tr>
<tr>
<td>tracking in the forward dynamic simulation of biomechanical multibody</td>
<td></td>
<td></td>
</tr>
<tr>
<td>systems</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Design of controller for single axis knee using MR damper</td>
<td>S. Seid, S. Sujatha and S. Chandramohan</td>
<td>989</td>
</tr>
<tr>
<td>Development of a human-robot dynamic model to support model-based</td>
<td>B. Ghannadi, N. Mehrabi and J. McPhee</td>
<td>999</td>
</tr>
<tr>
<td>control design of an upper limb rehabilitation robot</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Development of a musculotendon model within the framework of</td>
<td>A.R. Oliveira, S.B. Gonçalves, M. Carvalho and M.T. Silva</td>
<td>1008</td>
</tr>
<tr>
<td>multibody systems dynamics</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gait and posture responses to backpack load</td>
<td>M. Sayari, A. A. Yahmedi and N. Masmoudi</td>
<td>1018</td>
</tr>
<tr>
<td>Influence of muscle recruitment criteria on joint reaction forces</td>
<td>F. Michaud, U. Lugrís, Y. Ou, J. Cuadrado and A. Kecskemethy</td>
<td>1024</td>
</tr>
<tr>
<td>during human gait</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Inverse dynamics analysis of driver-seat-steering system using</td>
<td>M. Hada, E.I. Yasuda, T. Kobayashi, T. Yamaguchi and Y. Hattori</td>
<td>1032</td>
</tr>
<tr>
<td>multibody human model</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Modelling of sound transfer in middle ear with SMA element</td>
<td>R. Rusinek, J. Warminski and M. Szymanski</td>
<td>1044</td>
</tr>
<tr>
<td>On the effect of attachment position and compliance of wearable</td>
<td>J. Damerau, J. Jovic, T. Watanabe and U. Wolz</td>
<td>1056</td>
</tr>
<tr>
<td>robots on human joint and interface forces</td>
<td></td>
<td></td>
</tr>
<tr>
<td>predicted knee contact forces during walking</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Towards bridging the gap between motion capturing and biomechanical</td>
<td>T. Gail, R. Hoffmann, M. Miezal, G. Bleser and S. Leyendecker</td>
<td>1080</td>
</tr>
<tr>
<td>optimal control simulations</td>
<td></td>
<td></td>
</tr>
<tr>
<td>of acquired gait motions</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Railroad Vehicle Dynamics</td>
<td></td>
<td></td>
</tr>
<tr>
<td>detection</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
An innovative model-based dynamic weigh in motion system for railway vehicles
P. D'Adamio, A. Innocenti, L. Marini, E. Meli, L. Pugi and A. Rindi ........................................1113

Analysis of critical hunting speed and running safety of conventional railway vehicle truck on curved track
R.K. Sajeev and S. Chandramohan ........................................................................................................1125

Bifurcation analysis and hunting behavior of high-speed railway vehicle in a curve
M. Azarinezhad, M.R. Ghazavi and S. Rahmanian ..................................................................................1137

CFD assessment of the effect of windblown sand on a high-speed train
C. Paz, E. Suárez, C. Gil and M. Concheiro ............................................................................................1150

Computation of two-point contact between wheelset and rail exploiting the structure of non-penetrating contact equations for straight track
B. Fallahi and C. Pan ..................................................................................................................................1156

Computational model updating for a rail vehicle bogie system of a high-speed train based on combined component analysis
Q. Guo, Y. Tao, H. Li, D. Yang, L. Ma, B. Zhang, T. Wang and L. Zhang ..............................................1164

Methods to obtain reliable car body load assumptions for light rail vehicles using multibody simulations based on measured input
G. Zechel and M. Beitelschmidt .................................................................................................................1171

Modal Testing and Parameter Identification of an Overall Rail-vehicle Carbody of a High-speed Train for FE Model Validation
Q. Guo, D. Yang, H. Li, Y. Zhao, T. Wang, B. Zhang and L. Zhang ......................................................1183

Network-centric approach to adaptive real-time train scheduling
A.A. Belousov, P.O. Skobelev and M.E. Stepanov .....................................................................................1190

Numerical and experimental study on contact force fluctuation between wheel and rail considering rail flexibility and track condition
S. Ienaga, Y. Terumichi, K. Nishimura and M. Nishina ............................................................................1200

Partially-linearized multibody equations of railroad vehicles on arbitrary tracks for on-board applications
J. F. Aceituno, J. L. Escalona and D. García-Vallejo ..................................................................................1212

Road Vehicle Dynamics

A constraint embedding approach for complex vehicle suspension dynamics
A. Jain, C. Kuo, P. Jayakumar and J. Cameron .........................................................................................1221

A dynamic-based approach for road vehicle design: application to a three-wheeler
A. Verlé and P. Fisette .................................................................................................................................1236

A method to combine an MBD tire model with a thermo-dynamical one to improve the accuracy in the tire simulations
F. Calabrese, M. Baecker and A. Gallrein ..................................................................................................1248

An enhanced tire model for dynamic simulation based on geometrically exact shells
M. Roller, P. Betsch, A. Gallrein and J. Linn ............................................................................................1260

18
<table>
<thead>
<tr>
<th>Title</th>
<th>Authors</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Estimation of influence of stiffness of chassis frame on LCV dynamics in conditions of curvilinear motion</td>
<td>A.V. Tumasov, S.Y. Kostin, D.A. Butin, A.A. Vasiliev, A.A. Vashurin and Y.P. Trusov</td>
<td>1272</td>
</tr>
<tr>
<td>Estimation of manoeuvrability of multi-axis tractor-lorry-trailer combination for transportation of large-dimensioned indivisible loads</td>
<td>A.V. Tumasov, L.V. Barakhtanov, S.E. Manyanin, A.A. Vasiliev and S.L. Tropin</td>
<td>1279</td>
</tr>
<tr>
<td>Multibody and finite element models of a leaf-spring suspension for vehicle dynamics applications: numerical model, tests and correlation</td>
<td>A. Toso, U. Facchin and S. Melzi</td>
<td>1297</td>
</tr>
<tr>
<td>Parameter identification of a multibody vehicle model using mathematical optimization</td>
<td>J.L. Bueno-López, M.D. Gutiérrez-López and J. García de Jalón</td>
<td>1308</td>
</tr>
<tr>
<td>Robust high speed autonomous steering of an off-road vehicle</td>
<td>M. Kapp and P.S. Els</td>
<td>1310</td>
</tr>
<tr>
<td>Simulation of friction-induced vibrations using elastic multibody models</td>
<td>I. Iroz, P. Eberhard and M. Hanss</td>
<td>1318</td>
</tr>
<tr>
<td>Use of flexible models in extended Kalman filtering for vehicle body force estimation</td>
<td>S. van Aalst, F. Naets, J. Theunissen and W. Desmet</td>
<td>1329</td>
</tr>
<tr>
<td>Verification and optimization advantage of inerter devices apply to grounded vehicle dynamics</td>
<td>T.T. Tran and H. Hasegawa</td>
<td>1341</td>
</tr>
<tr>
<td>Vibration analysis of vehicle - UHPFRC wild bridge interaction</td>
<td>K. Nguyen Gia, O. Rio, M. Ralbovsky and B. Freytag</td>
<td>1353</td>
</tr>
<tr>
<td>Vibrations of a rolling tyre</td>
<td>I.F. Kozhevnikov</td>
<td>1365</td>
</tr>
<tr>
<td>Virtual articulated haulers - Simulation of vehicle performance and design loads</td>
<td>T. Ekevid, C. Carlqvist and T. Sjödén</td>
<td>1377</td>
</tr>
</tbody>
</table>

**Robotics**

<table>
<thead>
<tr>
<th>Title</th>
<th>Authors</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Development of an energy-saving manipulator using storage elements and reaction wheels</td>
<td>M. Iwamura, S. Imafuku, T. Kawamoto and W. Schiehlen</td>
<td>1388</td>
</tr>
<tr>
<td>Dynamic analysis using numerical multi-body approach for quadruped robots</td>
<td>M. D’Imperio, F. Cannella, F. Chen, L. Carbonari, D. Catelani and D.G. Caldwell</td>
<td>1400</td>
</tr>
<tr>
<td>Dynamic model of the PUMA-like manipulator with a hybrid position/force control in operation with a compliant environment</td>
<td>I.A. Orlov</td>
<td>1411</td>
</tr>
</tbody>
</table>
Dynamic parameters identification of a robot in simulated environment
A. A. Hayat, V. Abhishek and S. K. Saha .................................................................1417

Energy-efficient point-to-point trajectory generation for industrial robotic machines
P. Boscariol, G. Carabin, A. Gasparetto, N. Lever and R. Vidoni ................................1425

Exploiting the equations of motion for biped robot control with enhanced stability
J. Mayr, A. Reiter, H. Gattringer and A. Müller ......................................................1434

Modelling and control synthesis of flexible robot arm equipped with additional sensors
Z. Šika, L. Mráz, M. Valášek, J. Volech and P. Beneš .............................................1446

Multi-physics modelling of a compliant humanoid robot
A.A. Zobova, T. Habra, N. Van der Noot, H. Dallali, N.G. Tsagarakis, P. Fisette and R. Ronsse .............................................................................................................1458

On attitude dynamics and control of legged robots using tail-like systems
K. Machairas and E. Papadopoulos ..............................................................................1470

On the use of the wrench exertion capability as a performance index for cable driven robot
G. Boschetti and A. Trevisani ..................................................................................1482

Reduction of the effect of actuator saturation with periodic servo constraints
L. Bencsik, A. Zelei and L.L. Kovács ........................................................................1494

Sensitivity analysis on MIMBOT biped robot through parallel computing
A. Bustos, G. Carbone, H. Rubio, M. Ceccarelli and J. C. García-Prada ....................1504

Towards an enhanced controller for stable slope climbing
A.S. Mastrogeorgiou and E.G. Papadopoulos ..........................................................1516

Using the generalized inverted pendulum to generate less energy-consuming trajectories for humanoid walking
S. Omran, S. Sakka and Y. Aoustin .......................................................................1527

Benchmark Problems

A framework for data exchange and benchmarking of frictional contact solvers in multibody dynamics
C. Lacoursiere, M. Linde, Y. Lu and J. Trinkle ..........................................................1538

Benchmarking of augmented Lagrangian and Hamiltonian formulations for multibody system dynamics
F. González, D. Dopico, R. Pastorino and J. Cuadrado .............................................1548

Development of a flexible multibody simulation package for in-house benchmarking
M. Vermaut, T. Tamarozzi, F. Naets and W. Desmet .................................................1560
<table>
<thead>
<tr>
<th>Title</th>
<th>Authors</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Evaluation of the biomechanical simulator OpenSim on a multi-body system benchmark</td>
<td>L. Tagliapietra, M. Vivian, R. Caracciolo and M. Reggiani</td>
<td>1572</td>
</tr>
<tr>
<td>Multi-body dynamics benchmarks for frictionless elastic collisions</td>
<td>R.W. Bos and J. Wikkerink</td>
<td>1583</td>
</tr>
<tr>
<td>Testing the efficiency and accuracy of multibody-based state observers</td>
<td>E. Sanjurjo, J.L. Blanco, J.L. Torres and M.Á. Naya</td>
<td>1595</td>
</tr>
<tr>
<td>The Segway as a suitable multibody benchmark example</td>
<td>O. Verlinden, M. Hajzman, P. Polach and M. Byrtus</td>
<td>1607</td>
</tr>
<tr>
<td>Validation of flexible multibody dynamics beam formulations using benchmark problems</td>
<td>O.A. Bauchau, P. Betsch, A. Cardona, J. Gerstmayr, B. Jonker, P. Masarati and V. Sonneville</td>
<td>1619</td>
</tr>
</tbody>
</table>
A weak formulation for the numerical solution of the equations of motion of multibody systems subjected to scleronomic constraints

Nikolaos Potosakis, Elias Paraskevopoulos, Sotirios Natsiavas

Department of Mechanical Engineering
Aristotle University
54 124 Thessaloniki, Greece
nipotosa@auth.gr, eapcivil@gmail.com, natsiava@auth.gr

ABSTRACT
Some new theoretical and numerical results are presented on the dynamic response of a class of mechanical systems with equality motion constraints. At the beginning, the equations of motion of the corresponding unconstrained system are presented, first in strong and then in a weak form. Next, the formulation is extended to systems with motion constraints. The formulation is based on a new set of equations of motion, represented by a system of second order ordinary differential equations (ODEs) in both the coordinates and the Lagrange multipliers associated to the motion constraints. Moreover, the position, velocity and momentum type quantities are assumed to be independent, forming a three field set of equations. The weak formulation developed was first used to cast the equations of motion as a set of first order ODEs in the coordinates and the corresponding momenta. Then, the same formulation was also employed as a basis for producing a suitable time integration scheme for the systems examined. The validity and efficiency of this scheme was tested and illustrated by applying it to a number of characteristic example systems.

Keywords: Newton’s law of motion, Analytical dynamics, Motion constraints, Weak form.

1 INTRODUCTION
New results on the dynamics of constrained mechanical systems are of great interest to the engineering community since they are useful in many areas, including mechanisms, robotics, biomechanics, automotive and aerospace structures (e.g., [1-4]). Typically, the equations of motion for this class of systems are derived and cast in the form of a system of differential-algebraic equations (DAEs) of high index [3, 5, 6]. However, both the theoretical and the numerical treatment of DAEs is a delicate and difficult task [7]. For this reason, many attempts have been performed in the past in order to cure the problems related to a DAE modeling [8].

One way to avoid difficulties associated with the DAE formulation is to impose an appropriate scaling on the equations of motion and the generalized coordinates [9]. Another class of methods is based on differentiation of the constraint equations, reducing the index of the original DAEs to 2 or 1 [3, 5]. However, these methods need a stabilization since they lead to constraint violation in the lower kinematical levels [10]. Finally, another general category of methods is based on elimination of the Lagrange multipliers in order to convert the original DAEs to second order ODEs, by selecting a suitable set of independent generalized coordinates or velocities [6, 11]. This process is inconvenient to apply in practical problems. Moreover, other methods involving elimination of dependent velocities or the Lagrange multipliers do not possess an inherent mechanism to avoid constraint drift in the lower kinematical levels [12, 13].

The present formulation was based on a new set of equations of motion, represented by a coupled system of second order ODEs in both the coordinates of the system and the Lagrange multipliers associated to the motion constraints [14]. Here, these equations are first put in a convenient weak form. Moreover, the position, velocity and momentum type quantities were assumed to be independent, forming a three field set of equations [15-17]. The weak formulation developed was first used to cast the equations of motion as a set of first order ODEs in the coordinates and the corresponding momenta, resembling the classical Hamilton’s
canonical equations [18]. The same formulation can also be employed for producing suitable time integration schemes. One such scheme was developed for the purposes of this study. The validity and efficiency of this scheme was illustrated by applying it to a number of example systems. Among other things, the results obtained verify that the scheme developed passes successfully all the tests related to a special set of challenging benchmark problems, chosen by the multibody dynamics community [19].

The organization of this paper is as follows. First, the strong form of the equations of motion governing the dynamics of an unconstrained discrete mechanical system is presented briefly in the following section. Then, the corresponding weak form of these equations is derived in the third section. Next, the weak form of the equations of motion for the class of constrained multibody systems examined is derived and presented in the fourth section. Based on this form, the equations of motion are then derived as a set of first order ODEs in the generalized coordinates and the corresponding momenta. Finally, a temporal discretization scheme was also developed and numerical results were obtained for several characteristic examples. These results are presented in the fifth section. The work is completed by a summary of the main findings.

2 STRONG FORM OF NEWTON’S LAW ON A MANIFOLD

A class of mechanical systems whose position is determined by a set of generalized coordinates \( q^1, \ldots, q^n \), at any time \( t \), is examined [18]. The motion is represented by the motion of a fictitious point, say \( p \), along a curve \( \gamma = \gamma(t) \) in an \( n \)-dimensional manifold \( M \), the configuration space of the system [4, 20]. Moreover, the tangent vector \( \mathbf{v} = \frac{d\gamma}{dt} \) to this curve belongs to an \( n \)-dimensional vector space, the tangent space of the manifold at \( p \), denoted by \( T_p M \) [21].

For any point \( p \), a coordinate map \( \varphi \) can be defined, acting from a neighborhood of \( p \) on \( M \) to the Euclidean space \( \mathbb{R}^n \). Then, any point \( p \) on \( M \) is mapped through this map to a point

\[
q = \varphi(p)
\]

of \( \mathbb{R}^n \), with \( q = (q^1, \ldots, q^n) \) [22]. By adopting the usual summation convention on repeated indices [20], if \( c(t) \) represents a curve on \( \mathbb{R}^n \), its tangent vector at point \( q(t) \) of \( c(t) \) is

\[
\dot{q}(t) = \mathbf{G}_i(t) \dot{q}_i,
\]

where \( \mathcal{B}_g = \{ \mathbf{G}_1, \ldots, \mathbf{G}_n \} \) is a basis of \( \mathbb{R}^n \). As usual, one can define a natural (or holonomic, or coordinate) basis \( \mathcal{B}_g = \{ \mathbf{g}_1, \ldots, \mathbf{g}_n \} \) for space \( T_p M \) by

\[
\mathbf{g}_i = \varphi^{-1}_* \mathbf{G}_i, \quad i = 1, \ldots, n,
\]

where \( \varphi_*^{-1} \) represents the inverse of the tangent mapping \( \varphi_* \).

In Analytical Dynamics, it is frequently useful to employ a new basis \( \mathcal{B}_e = \{ \mathbf{e}_1, \ldots, \mathbf{e}_n \} \) of \( T_p M \) obtained by the holonomic basis \( \mathcal{B}_g \) through a convenient linear transformation

\[
\mathbf{e}_i = A_i^l \mathbf{g}_l, \quad L, i = 1, \ldots, n
\]

Then, any element \( \mathbf{u} \) of the vector space \( T_p M \) can be expressed in the equivalent forms

\[
\mathbf{u} = u^l \mathbf{e}_l = u^i \mathbf{g}_i,
\]

with
\[ u' = A'_i u^i \quad \text{and} \quad u'^i = B'^i u^i \] (6)

and

\[ B'^i A'_M = \delta^i_M \quad \text{and} \quad B'^i A'_L = \delta^i_J. \] (7)

Such a basis is characterized by its structure constants \( c^K_{IJc} \), defined through the Lie bracket by

\[ [\xi_i, \xi_j] = c^K_{IJc} \xi^K. \] (8)

In such cases, one can write

\[ u'^i = \tilde{\theta}^i \] (9)

but the quantities \( \theta^i \) may not be true coordinates. This, in conjunction with Eq. (6), leads to the concepts of quasi-coordinates and quasi-velocities [23]. Although the quasi-coordinates do not appear explicitly in the expressions used, their derivatives appear and are defined clearly in terms of the corresponding true coordinates. For instance, if \( f \) is a function on manifold \( M \), then

\[ \frac{\partial f}{\partial \theta^i} = \frac{\partial f}{\partial q^j} \frac{\partial q^j}{\partial \theta^i} = A'_i \frac{\partial f}{\partial q^j}. \] (10)

Then, the derivative of function \( f \) on manifold \( M \) along a vector field \( u \) can be determined by

\[ u(f) = f u' = f_{\mu} u^\mu. \] (11)

This means that one can simplify the notation, by dropping the upper case indices in the sequel, provided that it is clear what is the basis employed. For instance, the Lie derivative of a vector field \( u \) with respect to another vector field \( w \) on the configuration manifold \( M \) is given by

\[ \mathcal{L}_w u = [w, u] = (w^i u'_j - u^i w'_j + c^K_{ij} w^i u^K) \xi_i. \] (12)

Likewise, the covariant differential of \( u(t) \) along \( v \), has the form

\[ \nabla_v u(t) = \left( \dot{u}^k + A^k_{ij} v^i u^j \right) \xi_k, \] (13)

where \( \nabla \) is the affine connection of the manifold. A more common notation is the following

\[ \nabla_v u(t) = \frac{\partial u^k}{\partial t} \xi_k = u^k |^j v^j \xi_k, \]

with

\[ u^k |^j = u^k_j + A^k_{ij} u^i. \] (14)

The components \( \Lambda^i_{jk} \) of the connection \( \nabla \) in the basis of \( T_pM \), the affinities, are defined by

\[ \nabla_v \xi_j = \Lambda^k_{ij} \xi_k. \] (15)

Finally, one can define the dual space \( T^*_pM \), with elements known as covectors. In dynamics, a correspondence between a covector \( u^* \) and a vector \( u \) is established through the dual product

\[ u^*(w) = \langle u, w \rangle, \quad \forall w \in T_pM, \] (16)

where \( \langle \cdot, \cdot \rangle \) denotes the inner product of vector space \( T_pM \) [22]. In this way, to each basis \( \{ \xi_i \} \) (with \( i = 1, \ldots, n \)) of \( T_pM \), a dual basis \( \{ c^i \} \) can be established for \( T^*_pM \) by the condition
\[ \varepsilon_i^j (\xi_j) = \delta_i^j. \] (17)

Then, the covariant differential of a covector field \( \xi^i(t) \) on \( M \) along a vector \( \nu \) of \( T_p M \) is
\[ \nabla_{\nu} \xi^i(t) = (\dot{\nu}_k - \Lambda^i_k \nu^j \xi_j) \varepsilon^k. \] (18)

Determination of the true path of motion (or natural trajectory) on a manifold is based on application of Newton’s second law in the form
\[ \nabla_{\nu} \hat{p}^i = f^i, \] (19)
where \( f^i = f_i \varepsilon^i \) is the applied force [4, 24], while the generalized momentum is defined by
\[ \hat{p}^i \equiv \nu^i. \]

Then, if \( \nu = \nu^i \xi_i \) and \( \hat{p}_M^i = p_i \varepsilon^i \), application of Eq. (16) leads to
\[ p_i = g_{ij} \nu^j, \] (20)
where the quantities
\[ g_{ij} = \langle \xi_i, \xi_j \rangle \] (21)
represent the components of the metric tensor at point \( p \). These quantities are selected to coincide with the elements of the mass matrix \( G \) of the system, defined through the kinetic energy [20].

3 WEAK FORM OF NEWTON’S LAW ON A MANIFOLD

Through the definition of a class of special covectors (called Newton convectors, see [14]) by
\[ \hat{h}^i = \nabla_{\nu} \hat{p}^i - f^i, \] (22)
the equations of motion (19) at any point on a configuration manifold \( M \) can be put in the form
\[ \hat{h}^i = 0. \] (23)

Therefore, when there exist no motion constraints, it should be true that
\[ \hat{h}^i (w) = 0, \] (24)
for any element \( w \) of the tangent space \( T_p M \). This, in turn, implies that
\[ \int_{t_1}^{t_2} \hat{h}^i (w) dt = 0, \quad \forall w \in T_p M \] (25)
along a natural trajectory on the manifold and within any time interval \([t_1, t_2]\).

Manipulation of the last integral requires application of integration by parts of the covariant derivative appearing in Eq. (22). This is achieved by employing the relation
\[ \nabla_{\nu} (\hat{p}^i (w)) = (\nabla_{\nu} \hat{p}^i) (w) + \hat{p}^i (\nabla_{\nu} w), \] (26)
which can be interpreted as a Leibniz rule on differentiation [22]. Then, the following expression is obtained
\[ \int_{t_1}^{t_2} [\nabla_{\nu} (\hat{p}^i (w)) - \hat{p}^i (\nabla_{\nu} w) - f^i (w)] dt = 0. \] (27)
Finally, after an integration by parts of the first term inside the integral, the last equation becomes

\[ [P^*_M(w)]^0 - \int_{t_i}^{t_f} [P^*_M(\nabla_x w) + f^*_M(w)] dt = 0, \tag{28} \]

representing the weak form of the equations of motion [15, 16]. Namely, it constitutes an alternative form of the law of motion, as expressed by Eq. (19) originally.

Further manipulation of the weak form of the equations of motion (28) involves differentiations along the vectors \( v \) and \( w \). This task requires the construction of two smooth vector fields on the configuration manifold \( M \), based on these two vectors. The first of these vector fields can be constructed by considering the tangent vector \( v \) at each point of the natural trajectory \( \gamma(t) \). The second vector field can then be created by introducing another vector \( w \) of the tangent space at each point of the same trajectory, which can be arbitrary. Therefore, based on the above, a variation of any scalar function \( f \) is defined as the derivative of \( f \) along vector \( w \), by

\[ \delta f \equiv \dot{w}(f) = f_j w^j. \tag{29} \]

Likewise, the differential of \( f \) is defined by

\[ df \equiv \dot{v}(f) = f_j v^j. \tag{30} \]

Then, the variation and the differential of coordinate \( q^i \) are

\[ \delta q^i = q'^i w^j \quad \text{and} \quad dq^i = q'^i v^j. \tag{31} \]

4 **WEAK FORM OF NEWTON’S LAW FOR SYSTEMS WITH EQUALITY MOTION CONSTRAINTS**

Next, consider systems subject to an additional set of \( k \) scleronomic equality constraints

\[ \psi(q, v) = A(q) v = 0, \tag{32} \]

where \( v \) is a vector in \( T_p M \) and \( A = [a^R_i] \) is a known \( k \times n \) matrix. Considering each of the resulting scalar equations separately, these constraints can be viewed in the dual product form

\[ \psi^R(q, v) = (a^R_i(q))(v) = 0, \quad (R = 1, \ldots, k), \tag{33} \]

where the quantity \( a^R_i(q) \) represents the \( R \)-th row of matrix \( A(q) \). In the special case where the constraint is holonomic, this condition can be integrated and written in the algebraic form

\[ \phi^R(q) = 0. \tag{34} \]

Based on the above, the equations of motion of the systems examined can be put in the form

\[ h^*_M = h^*_C, \tag{35} \]

on the original manifold \( M \) [14], with

\[ h^*_M = h \dot{\xi} = [(g_{ij} v^j) - \Lambda^m_{ij}(g_m) v^j v^i - f_i] \dot{\xi} \tag{36} \]

and

\[ h^*_C = \sum_{R=1}^{k} [(\tilde{m}^{-1}_R \lambda^R) v^i + \tilde{c}^{-1}_R \lambda^R + \tilde{c}^{-1}_R \lambda^R - \tilde{f}_R \dot{\xi}^i], \tag{37} \]

where the convention on repeated indices does not apply to index \( R \). Moreover, the coefficients
\[ m_{R R} = c^R_i g_i / c^i_R, \quad \bar{c}_{R R} = -c^R_i \frac{\partial f}{\partial q^i}(q, v, t)c^i_R, \]
\[ k_{R R} = -c^R_i f_i,j(q, v, t)c^j_R. \]

(38)

are obtained through projections along a special direction \( \xi_R \) on \( T_p M \), specified by the action of the \( R \)-th constraint \([14]\). Specifically, the components of the \( n \)-vector \( \xi_R \) are selected from

\[ c^i_R \xi^i_R = 1. \]

(39)

These components are also needed for determining the projected forcing term

\[ \bar{f}_R = c^R_i f_i(q, v, t). \]

(40)

If generalized coordinates are used, Eq. (35) represents a set of \( n \) second order coupled ODEs in the \( n + k \) unknowns \( q^i \) and \( \lambda^R \). In addition, for each holonomic constraint, an ODE with form

\[ (m_{R R} \dot{\phi}^R)^{\prime} + \bar{c}_{R R} \dot{\phi}^R + k_{R R} \phi^R = 0 \]

(41)

is obtained, forcing both \( \phi^R \) and \( \phi^R \) to become zero eventually. Likewise, each nonholonomic constraint leads to

\[ (m_{R R} \psi^R)^{\prime} + \bar{c}_{R R} \psi^R = 0 \]

(42)

causing only \( \psi^R \) to become zero.

Taking into account the set of equations of motion (35), Eq. (25) is first modified accordingly to

\[ \int_t^\omega (\dot{h}^{\star}_w - \dddot{h}^{\star}_w)(w)dt = 0, \quad \forall w \in T_p M. \]

(43)

Next, for a holonomic constraint, expressed by Eq. (41), the following relation is satisfied

\[ \int_t^\omega \left[ (m_{R R} \dot{\phi}^R)^{\prime} + \bar{c}_{R R} \dot{\phi}^R + k_{R R} \phi^R \right] \delta \lambda^R dt = 0, \]

for an arbitrary multiplier \( \delta \lambda^R \). A similar expression is obtained for nonholonomic constraints.

In a weak formulation, it is frequently advantageous to consider the position, velocity and momentum variables as independent quantities \([17]\). For this, a new velocity field \( \nu \) is introduced on \( M \), which should eventually become identical to the true velocity field \( v \). This can be achieved by introducing a covector \( \pi \) as a set of Lagrange multipliers. A similar action can be taken for the velocity components \( \lambda^R \), by introducing another set of Lagrange multipliers, \( \sigma^R \). Likewise, one can relate the strong time derivatives \( v^i \) and \( \dot{\lambda}^R \) of the position type variables to weak velocities, denoted by \( \nu^i \) and \( \mu^R \), through two new sets of Lagrange multipliers, denoted by \( \delta \pi \) and \( \delta \sigma^R \), respectively. Finally, by adding up all these terms and performing appropriate mathematical operations yields eventually the following three field set of equations
\[ (p - \sum_{R=1}^{k} a_R^i \mathbf{m}_{RR} \mu^R - \pi_i) w^i + \sum_{R=1}^{k} (\mathbf{m}_{RR} \dot{\phi}_R^R - \sigma_R^i) \delta \lambda^R = 0, \]

\[ + \int_0^t \delta \pi_i (\nu^i - \nu^R) dt + \sum_{R=1}^{k} \int_0^t \delta \sigma_R^i (\mu^R - \lambda^R) dt \]

\[ + \int_0^t \tau^i = \int_0^t \left[ \frac{\partial}{\partial \nu^i} \left( \sum_{R=1}^{k} \left( \frac{\partial}{\partial \nu^i} \mathbf{m}_{RR} \dot{\phi}_R^R \right) \right) \right] \delta \lambda^R \] 

\[ + \sum_{R=1}^{k} \int_0^t (\sigma_R^i - \mathbf{m}_{RR} \dot{\phi}_R^R) \delta \lambda^R \] 

Equation (44) is convenient for performing an appropriate numerical discretization, leading to improvements in existing numerical schemes based on advanced analytical tools (e.g., [25, 26]). It can also be used for obtaining an alternative form for the equations of motion (35), expressed as a set of first order ODEs in the coordinate and the corresponding conjugate momentum variables. Specifically, by first collecting terms multiplied with \( \delta \pi_i \) yields

\[ \nu^i = \nu^j, \] 

which takes the more explicit form

\[ \dot{q}^i = \nu^i \quad \text{or} \quad \dot{A}^i \dot{\nu} = \nu^i \quad \Rightarrow \quad \dot{\theta} = B^j \nu^j, \]

for true or quasi-velocities are involved, respectively. Likewise, selecting terms of \( \delta \sigma_R^i \) leads to

\[ \lambda^R = \mu^R. \] 

Next, collecting terms multiplied by \( \delta \nu^j \) yields

\[ \pi_j = g_{ij} \nu^i - \sum_{R=1}^{k} a_R^i \mathbf{m}_{RR} \mu^R, \]

while selecting terms of \( \delta \mu^R \) leads to

\[ \sigma_R = \mathbf{m}_{RR} \dot{\phi}_R^R. \] 

The last two equations verify that the Lagrange multipliers \( \pi_i \) and \( \sigma_R^i \) appear as components of generalized momentum type quantities. Finally, collecting the terms multiplying \( w^i \) yields

\[ \frac{\partial \nu^i}{\partial \nu^i} = \int_0^t \left[ \frac{\partial}{\partial \nu^i} \left( \sum_{R=1}^{k} \left( \frac{\partial}{\partial \nu^i} \mathbf{m}_{RR} \dot{\phi}_R^R \right) \right) \right] \delta \lambda^R \]

while selecting terms of \( \delta \lambda^R \) leads to

\[ \dot{\sigma}_R = -\mathbf{m}_{RR} \dot{\phi}_R^R - \mathbf{f}_{RR} \phi^R. \] 

For a nonholonomic constraint, Eqs. (49) and (51) become

\[ \sigma_R = \mathbf{m}_{RR} \dot{\phi}_R^R \quad \text{and} \quad \dot{\sigma}_R = -\mathbf{m}_{RR} \dot{\phi}_R^R, \]
respectively.

The set of Eqs. (48) and (49) can be viewed as a linear algebraic system in the weak velocities \( \nu^i \) and \( \mu^R \). In principle, its solution furnishes these quantities as a function of the corresponding momentum quantities \( \pi_i \) and \( \sigma_R \). Consequently, substitution of these results in Eqs. (46), (47), (50) and (51) yields a system of first order ODEs involving the coordinates \( q^i \) and \( \lambda^R \) together with the conjugate momenta \( \pi_i \) and \( \sigma_R \) as unknowns. The resulting set of equations is expressed in the cotangent bundle \( T^*M \) and possesses a similar structure but is in fact more general than the form of the classical Hamilton’s canonical equations [18, 21]. Alternatively, direct substitution of Eqs. (48) and (49) into Eqs. (50) and (51) leads to a new system of first order ODEs involving the coordinates \( q^i \) and \( \lambda^R \) plus the velocities \( \nu^i \) and \( \mu^R \) as unknowns. Therefore, the resulting set of equations is now expressed in the tangent bundle \( TM \), instead.

5 NUMERICAL RESULTS

The weak form derived in the previous section is convenient for developing efficient numerical discretization schemes for the new set of equations of motion employed, leading to improvements over existing schemes (e.g., [5, 25, 26]). For the purposes of the present work, this form was first put within the framework of an augmented Lagrangian formulation [27-29]. This leads to a full exploration of the major advantages of the theoretical method applied, in a quite natural manner [14]. Specifically, this method is appropriate for performing a geometrically exact discretization. This is especially useful when the configuration space of the system possesses group properties [30]. The success of this formulation was demonstrated by the accurate solution obtained for a number of challenging problems. Some characteristic results are presented next.

5.1 Example 1: Planar Simple Pendulum

The first mechanical system examined consists of a sphere of unit mass, possessing a mass moment of inertia matrix equal to the \( 3 \times 3 \) identity matrix. The center of the sphere is mounted at one end of a massless rigid bar of unit length. The system rotates about a fixed axis Oy, as shown in Fig. 1a, with an initial angular velocity of -100 rad/s. In Fig. 1b is depicted the history of the angular velocity \( \omega_y \) over the first 30 seconds.

Next, results obtained by the present method are compared with those obtained by a state of the art code [31]. This code sets up the equations of motion as a system of DAEs and solves them by numerical integration. The results of the new method are labeled by LMD, while the results obtained by applying the stabilized index-1 technique of the code are labeled by SI1. Moreover, the numbers in the labels correspond to the error allowed in the calculations for each case. Quite similar results were also obtained by applying other DAE methods of the same code as well as by another state of the art code [32].

The results obtained by the DAE method deviate from the results of the new method, as shown in Fig. 1b. The discrepancy increases in a fast manner for solutions with smaller accuracy. Specifically, the magnitude of the angular velocity obtained by SI1 decreases with time, while the new approach captures the correct constant value. This deviation is reflected in other important quantities and is mostly due to the artificial damping involved in the calculations of the direct integration scheme employed by the code. For instance, a similar picture is obtained for the history of the kinetic energy of the system, as presented in Fig. 1c. Again, the new method captures the correct constant value of this quantity.
Figure 1. (a) A rigid sphere-bar system in pure rotation. Comparison of time histories of the: (b) angular velocity, (c) kinetic energy of the system.

5.2 Example 2: Rectangular Bricard Mechanism

The next set of results refers to a six-bar rectangular Bricard mechanism, shown in Fig. 2a. It belongs to a special set of benchmark problems [19]. All the rods are connected with revolute joints, have equal length and uniformly distributed mass. Again, this system moves due to gravity acting along the negative $y$-axis. The mechanism examined represents a mechanical system which is redundantly constrained throughout its motion.

In Fig. 2b are shown the time histories of the $x$, $y$ and $z$ coordinates of point $P_2$, while in Fig. 2c is depicted the mechanical energy of the system. Finally, in Figs. 2d and 2e are presented the corresponding histories of the constraint violations at the position and velocity levels during the same time interval, represented by the norm of the array of the constraints at each level.

Direct comparison of the results in Fig. 2 illustrates that the present method is accurate and passes successfully the benchmark tests. It also presents an improved numerical performance. For instance, the mechanical energy computed by the present method remains virtually constant (Fig. 2c). In addition, the errors in both the displacement and velocity constraint violations are bounded and stay at the same level, throughout the time interval examined (Figs. 2d and 2e).
Figure 2. (a) Mechanical model of a Bricard mechanism, (b) \( x, y \) and \( z \) coordinates of point \( P_2 \), (c) mechanical energy, (d) violation of position and (e) violation of velocity constraints.

5.3 Example 3: Andrews' Squeezer Mechanism

In the last mechanical example, shown in Fig. 3a, the planar, seven-link system known as Andrews’ squeezer mechanism is examined [19, 33]. A constant driving torque acts on rod OF during its motion. The system moves without gravity effects and its technical parameters are such that a relatively small time step is required during the numerical integration of the equations of motion.

In Fig. 3b are shown the time histories of the \( x \) and \( y \) coordinates of point F, while in Fig. 3c is depicted the balance of the mechanical energy of the system. Finally, in Fig. 3d is presented the corresponding history of the constraint violations at the velocity level. Again, the results illustrate the accuracy and efficiency of the present method.

Figure 3. (a) Mechanical model of Andrews’ squeezer mechanism, (b) \( x \) and \( y \) coordinates of point F, (c) mechanical energy balance, (d) violation of velocity constraints.
6 SYNOPSIS

In the first part of this work, a weak form of the equations of motion for a class of mechanical systems was derived. Specifically, in addition to obeying Newton’s law, the systems examined are subject to holonomic and/or nonholonomic scleronomic constraints. This formulation was based on a new set of equations of motion, represented by a coupled system of second order ODEs in both the generalized coordinates and the Lagrange multipliers associated to the motion constraints. Moreover, the position, velocity and momentum type quantities were assumed to be independent, forming a three field set of equations. The weak formulation developed was first used to cast the equations of motion to a set of first order ODEs in the coordinates and the corresponding momenta, resembling the structure of the classical Hamilton’s canonical equations. Then, it was used as a basis for producing a suitable time integration scheme for the class of systems examined. The numerical accuracy and efficiency of this new scheme was demonstrated by presenting results for a selected set of mechanical examples.

REFERENCES


[31] MotionSolve v11.0, User Guide, Altair Engineering Inc., Irvine, California, USA.


Adaptive meshing in two-dimensional beam elements based on the absolute nodal coordinate formulation

Antti I. Valkeapää, Aki M. Mikkola

Department of Mechanical Engineering
Lappeenranta University of Technology
Skinnarilankatu 34, 53850 Lappeenranta, Finland
[antti.valkeapaa, aki.mikkola]@lut.fi

ABSTRACT
The efficiency of finite element analysis relies not only on the correctly formulated finite elements and robust numerical solution but also on the reliable mesh creation and suitable mesh refinement. The absolute nodal coordinate formulation (ANCF) is a finite element based method developed in the multibody dynamics community to analyze structures undergoing large rotations and displacements. Substantial amount of research efforts have been directed towards creating a family of ANCF elements and applying these elements in various structure analyses. On the other hand, few studies have been conducted on the meshing in the absolute nodal coordinate formulation. However, different methods of mesh refinements for the accuracy enhancement of finite element solution at certain key locations have been studied in the finite element literature. The objective of this study is to analyze different adaptive meshing strategies for two-dimensional Euler–Bernoulli beam elements based on the absolute nodal coordinate formulation. More specifically, in this study investigation of goal-oriented and residual error based mesh refinement strategies is conducted.

1 INTRODUCTION
Multibody system dynamics (MSD) offers a straightforward computer-based approach to setting up and solving dynamic problems for complex mechanical systems. The multibody approach is rapidly becoming an essential development and diagnostic tool for a number of industries to address a wide range of engineering problems in areas such as robotics, aerospace applications, vehicle dynamics, biomechanics, and rotating structures [1]. Often in multibody system dynamics, the analyst assumes mathematically rigid interconnected bodies, an acceptable simplification for the analysis of motion and forces in many practical engineering problems. In an increasing number of cases, however, deformation must be considered to improve simulation accuracy. This deformation can be described using a number of formulation approaches [2]. Simpler approaches assume that body deformations are small and material behavior is elastic. This enables to utilize linear strain-displacements and linear stress-strain relations. More precise approaches consider how geometric changes due to deformation of a body influence system dynamic response by employing a nonlinear strain-displacement relation in the mathematical modeling. These approaches can also handle material nonlinearities using a nonlinear stress-strain relationship. Because the magnitude of body deformation varies greatly from component-to-component or place-to-place in a multibody system, it is common to combine the simpler and more advanced formulation approaches to achieve desired simulation accuracies with minimal computational burden. Many bodies can be treated as rigid, while some must be modeled with flexibility using a floating frame of reference formulation that assumes small elastic deformations.

Seeking to improve the handling of large body deformation in multibody applications, researchers have begun developing formulations to describe geometric and/or material nonlinearities. The most common is the large rotation vector formulation developed by the finite element community [3]. More recently, the multibody community has begun working on a second important approach, the absolute nodal coordinate formulation (ANCF). The absolute nodal coordinate formu-
lation (ANCF) is a finite element based method developed in the multibody dynamics community to analyze structures under going large rotations and displacements. This method was originally proposed by Ahmed Shabana 1996 [4]. Both the large rotation vector and absolute nodal coordinate formulations include features that make each most suitable for specific practical applications. To date, the more mature large rotation vector formulation has been implemented within several commercial finite element software products. The newer and still immature ANCF needs more studies before it can be widely implement to commercial products.

Growing continually since its inception in 1996, the ANCF has changed dramatically from its original form. Substantial amount of research efforts have been directed towards creating a family of ANCF elements and applying these elements in various structure analyses [5]. Recently, the origonal developer of the ANCF proposed clear conditions on the elements that can be condisered to be ANCF finite elements [6]. These conditions excluded the previously proposed $C^0$ continuity elements from the family ANCF elements. The efficiency of finite element analysis relies not only on the correctly formulated finite elements and robust numerical solution but also on the reliable mesh creation and suitable mesh refinement. Different methods of mesh refinements for the accuracy enhancement of finite element solution at certain key locations have been studied in the finite element literature [7]. On the other hand, few studies have been conducted on the meshing in the absolute nodal coordinate formulation. Yan et al. studied the effects of non-uniform mesh and distortion effects in a large deformation cantilever modeled with ANCF shell elements [8]. Hylr dahl et al. recently studied the effects of element distortions with thin ANCF plate elements [9]. Yu and Shabana utilized curvature nodes to define a thin ANCF plate element in such a way that interface nodes were converted to internal nodes with position degrees of freedom [10]. Valkeapää et al. conducted research on two different p-refinement strategies with two-dimensional Euler–Bernoulli ACNF beam elements [11].

The objective of this study is to analyze different adaptive meshing strategies for two-dimensional Euler–Bernoulli beam element based on the absolute nodal coordinate formulation. More specifically, in this study investigation of goal-oriented and residual error based mesh refinement strategies is conducted. The paper is organized into five chapters as follows. First, in Chapter 2, as an example of the kinematics of the ANCF beam element, a two-dimensional two-noded beam element is described followed by the derivation of the elastic energy using a material measure of curvature. Different adaptive mesh refinement strategies are described in the Chapter 3 and the numerical example and results are described in Chapter 4. The adaptive mesh refinement strategies are studied through a large deformation clamped-clamped beam subjected to point force numerical example. Finally, conclusions are drawn in Chapter 5.

2 KINEMATICS AND STRAIN ENERGY OF ANCF BEAM ELEMENT

In this section, the kinematics and elastic energy of the two-dimensional Euler–Bernoulli beam element based on the absolute nodal coordinate formulation will be explained briefly. In finite elements based on ANCF, kinematics can be expressed using spatial shape functions and global coordinates. Different sets of position and derivatives of position vector can be chosen as nodal degrees of freedom. The interpolation functions for the beam element used in this study are based on a polynomial of degree three; such as a set of base function $p_{1222} = \{1, x, x^2, x^3\}$. In the previous expression, the ANCF element is denoted using a recently proposed multidigit numerical code $dn\text{cm}$ [12]. In the multidigit numerical code, $d$ corresponds to dimension, $n$ is the number of nodes, $c$ is the number of nodal coordinates at the node, and $m$ is the vectorization multiplier showing the number of simultaneously interpolated fields. The kinematics of the beam element used in this study is shown in Figure 1.
This two-node beam element has a total of eight degrees of freedom as can be concluded from the multidigit code of the element 1222. Two degrees of freedom are for position and two are for position vector derivatives at each node. The position of an arbitrary particle in the beam element can be interpolated in the global fixed frame as follows:

\[ \mathbf{r} = \mathbf{S}_m(x)\mathbf{e}, \]

\[ \mathbf{e}^{(i)} = \begin{bmatrix} r_x^{(i)} & r_y^{(i)} \end{bmatrix}^T; \quad i = 1, \ldots, n \]

where the partial derivatives of first and higher order are expressed as follows:

\[ \mathbf{r}^{(i)} = \begin{bmatrix} r_x^{(i)} \\ r_y^{(i)} \end{bmatrix}, \quad \mathbf{r}^{(i)}_{,\alpha} = \frac{\partial^m \mathbf{r}^{(i)}}{\partial \alpha^m}; \quad \alpha = x^m \ldots \]

The shape functions can be presented using local normalized coordinates; such as \( \xi \in [-1 \ldots 1] \), as follows:

\[ S_1 = \frac{(2+\xi)(-1+\xi)^2}{4}, \quad S_2 = \frac{(1+\xi)(1-\xi)}{8}, \quad S_3 = \frac{(1+\xi)^2(2-\xi)}{4}, \quad S_4 = \frac{(1+\xi)^2(1-\xi)}{8}, \]

where relations \( \xi = 2x/l \), when the physical local coordinate system \( x \) is placed along the middle of the element. The shape functions can be represented in matrix form as:

\[ \mathbf{S}_m = \begin{bmatrix} S_1 & S_2 & S_3 & \ldots & S_{n \times c} \end{bmatrix} \]

where \( \mathbf{I} \) is a \( 2 \times 2 \) identity matrix, and \( n \times c \) is the number of shape functions defined using \( dncm \). The material measure of curvature is defined [13] as:
\[ \varepsilon_{xx}^1 = \frac{r_{1,x}r_{2,xx} - r_{2,x}r_{1,xx}}{\|r_x\|^2}, \]  
(5)

which is valid for planar problems only. The axial strain component of the Biot strain can be defined as:

\[ \varepsilon_{xx}^0 = \|r_x\|^2 - 1. \]  
(6)

The strain energy of one beam element of length \( l \) can be written as:

\[ W_{\text{int}} = \frac{1}{2} \int_0^l \left( EA \varepsilon_{xx}^0 + EI \varepsilon_{xx}^1 \right) dx \]  
(7)

where \( EA \) represents axial stiffness, and \( EI \) is bending stiffness. The vector of elastic forces can be defined as follows:

\[ \mathbf{q}_e = \frac{\partial W_{\text{int}}}{\partial \mathbf{e}} \]  
(8)

which are integrated using sufficient number of Gauss integration points.

3 ADAPTIVE MESH REFINEMENT STRATEGIES

The accuracy of the finite element numerical solution is dependent on the geometrical and material properties of the structure that is under investigation, its boundary conditions and the distribution of the loading. The error in the finite element solution can be diminished by mesh refinement. In this section, two adaptive mesh refinement strategies are described briefly. Two conventional methods to increase the accuracy of the finite element solution are p- and h-refinement. In the p-refinement the order of the basis function in the finite element is increased, for example, by means of additional element internal degrees of freedom. In the ANCF based finite elements this can be accomplished either by adding nodes with internal position degrees of freedom and/or additionally by utilizing higher order gradient degrees of freedom as nodal variables. In contrast, in the h-refinement the number of finite elements is increased while the order of the basis function is kept the same. This will reduce the solution error while the error is dependent on the element length [15]. It has been show by Babuska et al. that the combination of these two methods, the h-p refinement, leads to an exponential convergence rate [14]. The h-p refinement for the ANCF element is not investigated in this study. Rather, the h-refinement is used to seek improvement in the solution in conjunction with adaptive mesh refinement strategies.

Two adaptive mesh refinement strategies are investigated in this study: residual based mesh refinement and goal-orientated mesh refinement. The residual based mesh refinement utilizes a posteriori error criteria to determine whether the mesh is to be refined [15]. In contrast, the goal-oriented mesh refinement seeks to improve some specific field quantity in certain location. This can be for example displacement or rotation at some specific location in the structure under investigation. [16]. In this study, the error in the element total strain energy is used to determine if the mesh should be refined for the residual based mesh refinement strategy. The mesh refinement criteria is defined as:

\[ \varepsilon_{e}^{\text{re}} \geq 0.7 \times \varepsilon_{\text{max}}^{\text{re}} \]  
(9)

where \( \varepsilon_{i}^{\text{re}} \) is element \( i \) absolute total strain energy error with respect to reference solution and \( \varepsilon_{\text{max}}^{\text{re}} \) maximum absolute total strain energy error with respect to reference solution. If the element \( i \) strain energy error fulfills the criteria shown in Equation 9 then the element is split into two
elements in the subsequent analysis. The goal-orientated strategy is harnessed to improve the displacements at specific location that is explained more detail in the numerical example section. The mesh refinement criteria for the goal-orientated strategy is defined as:

\[ \varepsilon_{ge}^i \geq 0.7 \times \varepsilon_{ge}^{max} \]  

(10)

where \( \varepsilon_{ge}^i \) is element \( i \) absolute displacement error with respect to reference solution and \( \varepsilon_{ge}^{max} \) maximum absolute displacement error with respect to reference solution. If the element \( i \) displacement fulfils the criteria shown in Equation 10 then the element is split into two elements in the subsequent analysis.

4 NUMERICAL EXAMPLE

In this section, clamped-clamped beam with a rectangular cross-section under point force shown in Figure 2 is studied as numerical example to demonstrate the differences between conventional, residual based and goal-oriented mesh refinement strategies. The geometrical and material parameters of the beam are: length \( L = 2 \) m, height \( H = 0.1 \) m, width \( W = 0.1 \) m, and Young’s modulus \( E = 2.07 \cdot 10^{11} \) N/m². The vertical point load is set to act on the beam at \( L_f = 0.5 \) m with magnitude \( F_e = \frac{1250EI}{L^2} \) N leading to large deformation. For the sake of simplicity, linear stress-strain material behavior is employed as the purpose of the numerical example is to investigate mesh strategies. At the clamped ends, \( r_{1,x} \) nodal coordinates are left free and all other nodal coordinates are fixed.

Adaptive refinement strategies used in this numerical example are uniform h-refinement, residual strain energy error and goal-oriented mesh refinement. The numerical example is solved using Maple 8 commercial symbolic computation software capable of solving numerical problems with an arbitrary number of digits with similar computer code implementation as earlier study on the h- and p-refinement in ANCF elements [11]. In the numerical example, 30 digits are used in the numerical calculation. The tangent stiffness matrix is determined using a five-point finite element formulation. The off-balance force L2-norm convergence criteria of \( 1 \times 10^{-16} \) is used in the Newton iteration. The number of Gauss integration points is defined such that the exact integration is fulfilled significantly enough for the purpose of convergence analysis. The convergence of each mesh analysis was obtained using nine iterations with full Newton’s method. The rate of convergence (ROC) numerical values are calculated using a linear fit in a double logarithmic scale. Convergence of the absolute displacement error in \( u_x \) and \( u_y \) at the force application point \( L_f \) with respect to the reference result calculated using a mesh of uniformly distributed 32 ANCF beam elements are shown.
in Figure 3 and the absolute error of the total strain energy in Figure 4. All of the methods used started with the same uniform four element mesh but ended in different mesh of eight elements. The uniform h-refinement refined the mesh to the eight exactly same length elements. The residual error based method first refined the mesh to consist of five elements with one element length halved next to external load application point followed by mesh eight elements. The goal-oriented method first refined the mesh to consist of six element with two of the elements near external load application point halved followed by mesh of eight elements.

Figure 3. Absolute error of displacements at force application point with different refinement strategies of clamped-clamped beam under point force.

Figure 4. Absolute error of total strain energy with different mesh refinement strategies of clamped-clamped beam under point force.
5 CONCLUSIONS

In this study, meshing strategies for beam elements based on the absolute nodal coordinate formulation were investigated. To this end, goal-oriented and residual error based mesh refinement strategies were compared against conventional uniform h-refinement. This study concluded that the adaptive mesh refinement strategies for the clamped-clamped beam subjected to point force displacements were able to outperform the uniform h-refinement but not consistently. The residual error based method that utilized the absolute error of the strain energy within each element as mesh refinement criteria lead to the best rate of convergence in the absolute error of the total strain energy and in the absolute error of the displacement in the y-coordinate direction. However, the uniform h-refinement was able to diminish the absolute error of the displacement in the x-coordinate direction more efficiently than the residual error based method. The goal-oriented method was the most efficient in improving the displacement in the x-coordinate direction.

FUNDING

The authors thank the Academy of Finland for funding (application No 285064)

REFERENCES


An Approach to the Parameterization of Dynamic Models of Serial Mechanisms

Sergey Krutikov

Department of Robotics and Mechatronics
Bauman Moscow State Technical University
Baumanskaya 2-ya str., 5, 105005 Moscow, Russia
sergey.krutikov@bmstu.ru

ABSTRACT

Canonical motion equations are used for performing some analysis of a multibody system as well as for implementing an efficient code which solves direct and inverse dynamics. Using classical parameterization, the Lagrangian function which generates these equations is completely determined by numerous generally nonlinear combinations of geometrical and inertial parameters of a mechanism. However, there has been no universal approach to direct calculation of these combinations so far. Taking into account some features of the structure of the Lagrangian function this paper suggests such an approach for serial multibody systems. A new set of parameters referred to as generalized inertial parameters is introduced. Simple relations for direct calculation of all constant coefficients of canonical equations using this set are obtained. A part of the generalized parameter set which constitutes the minimal parameter set of mechanism dynamics is found. Applications of the suggested approach to generation of the motion equations and of the identification model are presented. An example demonstrating its usage is also included.

Keywords: multibody, dynamic, symbolic, model, parameter.

1 INTRODUCTION

Canonical motion equations of multibody systems (mechanisms) are used for modeling their motion and development of control algorithms. Very often these tasks must be implemented in real time, which requires high performance. One way to achieve this is to use general-purpose numerical schemes, and another one is to generate the model symbolically. When applied to a particular mechanism, the latter approach is considered to be several times faster than the former [1]. The same is true for the identification model of a mechanism.

Since the structure of the canonical equations is well known, a set of constant coefficients must be computed to determine them. These coefficients are some generally nonlinear combinations of geometrical and inertial parameters of a mechanism. However, a universal approach to straightforward calculation of these combinations is still an open issue. Usually regular algorithms of the numerical model formulation in conjunction with general-purpose or specialized computer algebra systems are used to obtain expressions of the equations and perform some simplifications [1, 2]. All such software packages (e.g., SyMoRo [3]) do take into account the details of rigid body kinematics and dynamics but do not pay attention to a predefined structure of the Lagrangian function which actually generates the canonical equations.

In this paper, another approach to parameterization of the dynamic model of serial mechanisms is presented. The paper is organized as follows. At first, some main concepts of serial mechanism dynamics are recalled. Then, a theorem giving a theoretical basis of the suggested approach is formulated and proved. A new parameterization of mechanism dynamics – the set of generalized inertial parameters – is introduced as a corollary of this theorem. The representation of canonical motion equations using this set is obtained and simple relations for direct calculation of their coefficients are derived. After that, the minimal parameter set of mechanism dynamics is found as a certain subset of the generalized parameter set and formulation of the energy identification
model in terms of this subset is presented. Then, some application notes to symbolical generation of the motion equations are given. Finally, the approach is demonstrated for a general planar robot with two revolute joints.

2 CLASSICAL FORMULATION OF LAGRANGIAN DYNAMICS

Consider an open serial kinematic chain of \( N \) rigid bodies connected with lower pair joints that provide \( n \geq N \) degrees of freedom (DoF). Only the joints which are kinematically equivalent to a chain of rotational and translational ones are allowed. Thus, the total number of links is \( n \) and \( n - N \) links are virtual. In that case, we can consider all joints as either rotational or translational. Let us attach the moving frame \( k \) to the link \( k \) (\( k = 1, \ldots, n \)) with respect to classical Denavit-Hartenberg notation and attach fixed world frame 0 to the base of a mechanism. Then the transition from frame \( k-1 \) to frame \( k \) can be expressed by a homogenous transformation matrix

\[
{k-1}^k (q_k) = 
\begin{pmatrix}
\cos \theta_k & -\sin \theta_k \cos \alpha_k & \sin \theta_k \sin \alpha_k & a_k \cos \theta_k \\
\sin \theta_k & \cos \theta_k \cos \alpha_k & -\cos \theta_k \sin \alpha_k & a_k \sin \theta_k \\
0 & \sin \alpha_k & \cos \alpha_k & d_k \\
0 & 0 & 0 & 1
\end{pmatrix},
\]

(1)

where

- \( \theta_k \) is the angle between \( x_{k-1} \) and \( x_k \) rotating around \( z_{k-1} \),
- \( \alpha_k \) is the angle between \( z_{k-1} \) and \( z_k \) rotating around \( x_k \),
- \( a_k \) is the distance between \( z_{k-1} \) and \( z_k \),
- \( d_k \) is the distance between \( x_{k-1} \) to \( x_k \).

Here, \( x_k \) and \( z_k \) are unit vectors of the respective coordinate axes of link frames. Generalized coordinate \( q_k \) of link \( k \) is \( \theta_k \) if joint \( k \) is rotational and \( d_k \) if it is translational. The transition from the world frame to frame \( k \) is determined by the following transformation matrix

\[
{0}^k (q) = 0^1 (q_1) \cdots {k-1}^k (q_k),
\]

(2)

where \( q \) is a generalized coordinates vector.

Canonical motion equations of such mechanisms are usually derived from the Euler-Lagrange equations

\[
dt \frac{\partial L}{\partial \dot{q}_k} - \frac{\partial L}{\partial q_k} = Q_k, \quad k = 1, \ldots, n,
\]

where \( L \) is the Lagrangian function defined as the difference between kinetic and potential energy and \( Q_k \) are non-potential generalized forces. It is well known that the Lagrangian \( L \) for a non-screw non-oscillating \( n \) DoF mechanical system like a serial mechanism is a quadratic function of generalized velocities \( \dot{q}_i \):

\[
L = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n a_{ij} (q) \dot{q}_i \dot{q}_j - P(q).
\]

(3)

Here, \( a_{ij} \) is a symmetric positive-definite tensor field of generalized inertia and \( P \) is a scalar field which represents the potential energy of gravity. This results in the following equations of motion [5]:

\[
\sum_{i=1}^n a_{ki} (q) \ddot{q}_i + \sum_{i=1}^n \sum_{j=1}^n b_{kij} (q) \dot{q}_i \dot{q}_j + c_k (q) = Q_k, \quad k = 1, \ldots, n,
\]

(4)

where \( \ddot{q}_i \) are generalized accelerations, \( b_{kij} = \left( \partial a_{ki} / \partial q_j + \partial a_{kj} / \partial q_i - \partial a_{ij} / \partial q_k \right) / 2 \) are Christoffel symbols of the first kind for generalized inertia tensor and \( c_k = \partial P / \partial q_k \) are generalized forces.
due to gravity. Left-hand side of the latter equation determines own dynamics of a mechanism, whereas generalized forces $Q_k$ describe external actions. They usually include joint drive torques and reactions of non-ideal constraints, e.g. joint friction.

There are a lot of ways for calculating the components of generalized inertia and potential energy. In this paper, $4 \times 4$ matrices approach [4] is used:

$$d_{ij} = \sum_{k=\max(i,j)}^{n} \text{tr}(U_{ki}H_kU_{kj}^T),$$

$$P = - (g^T 0) \sum_{k=1}^{n} 0^T H_k e_4.$$  

Here, $g$ is the acceleration of gravity defined with respect to the world frame, $e_i$ is a unit vector which $i$-th element is 1 and $H_k$ is $4 \times 4$ inertia matrix of link $k$ with respect to its attached frame:

$$H_k = \begin{bmatrix} I_{xx}^k & I_{xy}^k & I_{xz}^k & S_x^k \\ I_{yx}^k & I_{yy}^k & I_{yz}^k & S_y^k \\ I_{zx}^k & I_{zy}^k & I_{zz}^k & S_z^k \\ S_x^k & S_y^k & S_z^k & m_k \end{bmatrix}.$$  

It comprises all inertial parameters of the link: mass $m_k$, first moments $S_k^u$ and second moments $I_{uv}^k$, $u, v \in \{x, y, z\}$. Inertia matrix of virtual links is zero matrix. In this paper, the components of $H$ are referred to as the classical inertial parameters. Matrices $U_{ki}$ are partial derivatives $\partial T_k/\partial q_i$ and can be calculated as follows

$$U_{ki} = \begin{cases} 0, & k < i, \\ 0^T_{1} \ldots D_{k-i-1}^T T_{i-k-1} \ldots 0^T, & k \geq i, \end{cases}$$

where $D_i$ is a constant matrix of a predefined form.

3 ALTERNATIVE APPROACH

The suggested approach is based on the interpretation of the Lagrangian function not as a tensor field but as a vector in the linear space of all continuous quadratic functions of the same form. Generally, this space is infinite dimensional since the coefficients of the Lagrangian are arbitrary functions of generalized coordinates $q$. However, due to the special structure of generalized inertia and of potential energy it is possible to find a finite dimensional linear subspace which definitely contains the Lagrangian $L$. The following theorem gives theoretical foundation of the approach being presented.

Theorem 1. The potential energy of gravity $P$ and the components of generalized inertia tensor $a_{ij}$ of a serial mechanism belong to a finite dimensional linear space $B$ constructed as a tensor product of elementary subspaces $F_k$

$$B = F_1 \otimes \ldots \otimes F_n,$$

which are spanned by the following sets of basis functions$^2$:

$$F_k = \{ 1(q_k), \cos q_k, \sin q_k, q_k, \cos(2q_k), \sin(2q_k), q_k^2 \}.$$  

$^1$Moments $I_{uv}^k$ are not moments of inertia about any coordinate axis, but are connected with them uniquely via a linear map.

$^2$Hereinafter, $1(x)$ means function which equals to 1 for all $x$ from its domain.
Proof. At first, we need to show that if the operation \( \otimes \) is a regular scalar multiplication, space \( \mathbb{B} \) exists, i.e. it is spanned by the set of all tensor products of elementary basis functions [6]. Consider two finite dimensional linear spaces of scalar functions of different generally non-scalar variables. Let \( \{g_i(x), i = 1, \ldots, l\} \) and \( \{h_j(y), j = 1, \ldots, m\} \) be the basis sets of these spaces. Suppose the set \( B = \{g_i h_j, i = 1, \ldots, l, j = 1, \ldots, m\} \) of tensor products of basis functions is linearly dependent. Then we can find a set of coefficients \( c_{ij} \), not all zero, satisfying the following condition

\[
\sum_{i=1}^{l} \sum_{j=1}^{m} c_{ij} h_j g_i = 0
\]

for all \( x \) and \( y \) from their domains. Since \( h_j \) are linearly independent and not all of \( c_{ij} \) are zero, there exists such a point \( y^* \) that at least one linear combination \( \sum_{j=1}^{m} c_{ij} h_j(y^*) \) out of \( l \) is not zero. This fact means the set of \( g_i \) is linearly dependent. The latter contradicts the assumption that they constitute a basis. Therefore, the set \( B \) is linearly independent and spans a tensor product of the respective spaces. Applying this result to spaces \( \mathbb{B}_k (k = 2, \ldots, n) \) recursively, where \( \mathbb{B}_k = \mathbb{B}_{k-1} \otimes \mathbb{B}_k \) and \( \mathbb{B}_1 = \mathbb{F} \), we show that the space \( \mathbb{B} = \mathbb{B}_n \) exists and the set \( \{f_1 \ldots f_n : f_k \in F_k, k = 1, \ldots, n\} \) is its natural basis.

Now we must prove that \( P \) and \( a_{ij}(q) \) belong to \( \mathbb{B} \). It is clear that Equation 1 can be represented in the following form

\[
k^{-1}T_k(q_k) = \sum_{l=1}^{7} A_{l} T_k(q_k),
\]

where \( f_k(q_k) \in F_k \) and \( A_{l} \) are constant matrices which depend on geometrical parameters of link \( k \). Only matrices \( A_{1}^l, A_{2}^l, A_{3}^l \) are nonzero if joint \( k \) is rotational and matrices \( A_{4}^l, A_{5}^l, A_{6}^l \) if it is translational. Let us define a set of tensor indices \( I^k \) as a \( k \)-th cartesian power of the set \( I = \{1, \ldots, 7\} \) and denote the product \( f_1 \ldots f_n \) by \( b_n^l \), where \( l = (l_1, \ldots, l_n) \in P'. \) Then

\[
0T_k(q) = \sum_{l \in P'} T_k b_n^l(q),
\]

where \( T_k = A_{1} \ldots A_{7} \) if \( l = (l_1, \ldots, l_1, 1, \ldots, 1) \) and \( l_m \leq 4 (m = 1, \ldots, k) \) and \( T_k = 0, \) otherwise. Taking this into account, potential energy can be represented as follows

\[
P(q) = \sum_{l \in P'} \mu_l b_n^l(q), \tag{7}
\]

where

\[
\mu_l = -(g^T 0) \sum_{k=1}^{n} T_k H_k e_4 \tag{8}
\]

Similarly to Equation 3, we have

\[
U_{kl} = \sum_{l \in P'} U_{kl} b_n^l(q),
\]

where \( U_{kl} = A_{1}^l \ldots D_{l} A_{l_k} \ldots A_{k}^l \) if \( k \geq i, \ l = (l_1, \ldots, l_k, 1, \ldots, 1) \) and \( l_m \leq 4 (m = 1, \ldots, k) \) and \( U_{ki} = 0, \) otherwise. Consider the matrix product \( U_{kl} U_{ki} \). Using the latter equation and recalling that functions \( b_n^l \) are products of elementary basis functions \( f_k^l \), it can be written in the following form

\[
U_{kj} U_{ki} = \sum_{m \in P'} \sum_{l \in P'} \left[U_{kj}^m U_{ki} \right] f_1^{m_1} f_1^{m_2} \ldots f_4^{m_4}.
\]

We can consider that all elements of tensor indices \( l \) and \( m \) are less than 5 since matrices \( U_{kl} \) and \( U_{kj} \) are zero, otherwise. It can be easily checked that all products \( f_k^l f_k^{m_k} (l_k, m_k = 1, \ldots, 4) \) are
linear combinations of the elements of \( F_k \). Therefore,

\[
U^\top_{kj_i} U_{ki} = \sum_{r \in P} \hat{U}_{kj_i}^r f_i^r \cdots f_n^r,
\]

where matrices \( \hat{U}_{kj_i}^r \) are some linear combinations of \( \left(U_m^w \right)^\top U^l_{kj_i} \). Thus,

\[
a_{ij}(q) = \sum_{l \in F} \mu^l_{ij} \tilde{b}^l_n(q), \tag{9}
\]

where

\[
\mu^l_{ij} = \sum_{k = \max(i,j)}^n \operatorname{tr} \left( \hat{U}_{kj_i}^l H_k \right). \tag{10}
\]

Since the set of all \( \tilde{b}^l_n \) constitutes the natural basis of space \( \mathbb{B} \), Equations 7 and 9 mean that the potential energy and the components of generalized inertia belong to this space.

### 3.1 Generalized inertial parameters

Applying Theorem 1 to Equation 3, we obtain that the Lagrangian \( L \) belongs to a linear space

\[
\mathbb{B}_L = F_0 \otimes \mathbb{B},
\]

where

\[
F_0 = \operatorname{span} \left\{ \{1(q)\} \cup \{\tilde{q}_i^2/2, i = 1, \ldots, n\} \cup \{\tilde{q}_i \tilde{q}_j, i = 2, \ldots, n, j = 1, \ldots, i - 1\} \right\}.
\]

Therefore, the set of all coefficients \( \mu^l_{ij} \) and \( \mu^l_{ij} (i = 1, \ldots, n, j = 1, \ldots, i) \) generates the set of coordinates of the Lagrangian in the natural basis of \( \mathbb{B}_L \). Let us define this set as the set of \textit{generalized inertial parameters} of a mechanism since these coordinates are some linear combinations of inertial parameters of the links. They determine dynamics of a mechanism completely and uniquely according to the properties of bases. Technically, total amount of generalized inertial parameters equals to the dimension of space \( \mathbb{B} \), which is extremely high: \( \dim \mathbb{B} = \left(1 + (n^2 + n)/2\right)^7 n \). However, a lot of generalized parameters are actually zero apriori. The issue of reduction of the generalized parameter set is analyzed later in the paper.

### 3.2 Motion equations

Now we should obtain the motion equations of a mechanism in terms of generalized inertial parameters. According to Equation 4, the derivatives of generalized inertia components \( a_{ij} \) and of potential energy \( P \) with respect to generalized coordinates \( q_k \) must be calculated. It obviously leads to calculating similar derivatives of basis functions \( \tilde{b}^l_n \) with respect to Equation 7 and Equation 9. Fortunately, space \( \mathbb{B} \) is closed under differentiation operation \( \partial / \partial q_k \). Indeed,

\[
\frac{\partial \tilde{b}^l_n}{\partial q_k} = f_i^l \cdots \frac{\partial f_k^l}{\partial q_k} \cdots f_n^l.
\]

It can be easily checked that each space \( F_k \) is closed under differentiation with respect to \( q_k \), i.e.

\[
\frac{\partial f_i^l}{\partial q_k} = \alpha^l(q_k) f_i^\eta(l), \quad \forall l \in I.
\]

The functions \( \alpha^l(q_k) \) and \( \eta^l(q_k) \) map the set \( I \) to the set \( C = \{0, \pm 1, \pm 2\} \) of integers and to itself, respectively. They are defined in Table 1.

---

3 Mixed products \( \tilde{q}_i \sin \tilde{q}_k \) and \( \tilde{q}_i \cos \tilde{q}_k \) are not taken into account since their coefficients in the resulting combination are zero apriori. This is due to the fact that screw joints are not allowed, so matrix \( A^2_k \) can not be nonzero if matrices \( A^2_l \) and \( A^2_k \) are not and vice versa.
Substitution of the latter equation into Equation 7 and Equation 9 gives us

Applying this result in conjunction with Equation 9 to the motion equations 4, we obtain

The definition of \( \eta (l) \) implies that it is bijective and, hence, so are all the functions \( \eta_k (l) \). This means they are just permutations of tensor indices, i.e. if we take sequentially all the indices \( l \in I^n \), we do the same for \( \eta_k (l) \in I^n \) but in a different order. Therefore, we can rewrite the last two equations as follows:

Applying this result in conjunction with Equation 9 to the motion equations 4, we obtain

where the constant coefficients \( a_{kl}^l \), \( b_{klj}^l \) and \( c_k^l \) are calculated as follows

This means that some coefficients of motion equations are exactly the generalized inertial parameters (up to a multiplier) and other ones are linear combinations of them.

### 3.3 The minimum parameter set

The set of generalized inertial parameters completely determines the Lagrangian function and, therefore, dynamics of a mechanism. However, the number of these parameters is enormous, which makes it difficult to use them in real applications. To make reduction of the parameter set

### Table 1. The definition of the functions \( \alpha (l) \) and \( \eta (l) \).

<table>
<thead>
<tr>
<th>( l )</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \alpha (l) )</td>
<td>0</td>
<td>-1</td>
<td>1</td>
<td>1</td>
<td>-2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>( \eta (l) )</td>
<td>7</td>
<td>3</td>
<td>2</td>
<td>1</td>
<td>6</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>( \eta^{-1} (l) )</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td>7</td>
<td>6</td>
<td>5</td>
<td>1</td>
</tr>
</tbody>
</table>
we should exclude the parameters equal to zero and find out if there is a dependency between different generalized parameters.

To work out this problem, consider a linear space $\mathbb{P}$ of all classical inertial parameters which is isomorphic to $\mathbb{R}^{10N}$. The generalized inertial parameters are elements of this space since they linearly depend on the classical ones which constitute the standard basis of $\mathbb{P}$. Thus, the connection between these two parameter sets is determined by a linear mapping

$$\boldsymbol{\mu} = C \boldsymbol{p},$$

where $\boldsymbol{\mu}$ is the column vector of all generalized parameters, $\boldsymbol{p} \in \mathbb{P}$ is the column vector of all classical parameters and $C$ is a matrix of the respective size which depends only on the types and parameters of joints. As this matrix is calculated symbolically, generalized parameters which correspond to its zero rows are definitely zero. Additional simplification for a particular mechanism can be performed by discovering the row subspace which is orthogonal to the given classical parameters vector. In order to calculate the elements of $C$, we can give unity value to one of the classical parameters and zero values to the others. Then, the respective column of matrix $C$ is equal to the value of vector $\boldsymbol{\mu}$. It is equivalent to calculating partial derivatives of the generalized parameters with respect to the classical ones. Denoting the $i$-th classical inertial parameter of link $k$ by $p^i_k$, we have

$$\frac{\partial \mu^l_k}{\partial p^l_k} = -(g^T 0) T^l_kD_H e_4,$$

$$\frac{\partial \mu^l_{ij}}{\partial p^l_k} = \begin{cases} \text{tr} \left( \tilde{U}^l_{k,i,j}D_H \right), & k \geq \max(i, j), \\ 0, & \text{otherwise} \end{cases},$$

where $DH^l_k = \partial H_k / \partial p^l_k$ is a constant matrix whose elements are zero, except one or two which are unity. Also, matrices $DH^l_k$ are zero if the link is virtual.

Once matrix $C$ has been calculated and its rows corresponding to zero parameters have been excluded, the next step of the parameter set reduction can be performed. Let us consider the transpose of $C$ to be the coordinate matrix of nonzero generalized inertial parameters in the standard basis of $\mathbb{P}$. Then, the linearly independent rows of $C$ determine the basis parameters spanning a linear subspace of $\mathbb{P}$ which contains the whole set of generalized inertial parameters. The dimension $r$ of this subspace equals to rank $C \leq \min(10N, \dim \mathbb{B})$. Let $P$ be such a row permutation matrix that

$$PC = \begin{pmatrix} C_1 \\ C_2 \end{pmatrix},$$

(15)

where rank $C_1 = r$. Therefore, the vector of basis generalized parameters $\boldsymbol{\mu}_b$ and the vector of the other ones $\boldsymbol{\mu}_d$ are expressed through the classical parameters $\boldsymbol{p}$ as follows:

$$\boldsymbol{\mu}_b = C_1 \boldsymbol{p},$$

$$\boldsymbol{\mu}_d = C_2 \boldsymbol{p}.$$

Also the dependent parameters $\boldsymbol{\mu}_d$ can be represented as linear combinations of the basis ones:

$$\boldsymbol{\mu}_d = C_\mu \boldsymbol{\mu}_b,$$

(16)

where $C_\mu$ is a coordinate matrix of the respective dimensions. Due to linear independence of the elements of $\boldsymbol{p}$, the last three equations result in $C_\mu C_1 = C_2$. The solution with respect to $C_\mu$ is $C_\mu = C_2 C_1^+$, where "$+$" means Moore-Penrose pseudoinverse.

4The set of classical parameters of the link is supposed to be ordered as follows $\{I_{xx}, I_{yy}, I_{zz}, I_{xy}, I_{xz}, I_{yz}, S_{x}, S_{y}, S_{z}, m_{k}\}$
Thus, the set of all generalized inertial parameters can be reduced to the set of \( r \) basis ones which still completely determine the dynamic model. This is due to the fact that basis set spans the subspace of all generalized inertial parameters of a mechanism. Indeed, zero parameters do not affect the model and the others are expressed via the basis ones. The basis parameter set is also minimal since elimination of any parameter from this set implies that it does not span the subspace of generalized parameters anymore. This concept is very close to the one presented in [7]. However, the connection between the constant coefficients \( a_{kl}, b_{kij}, c_{k} \) of motion equations and the minimal parameter set is not clear from this work.

### 3.4 The identifiable parameter set

It is well known in multibody dynamics that not all inertial parameters of a mechanism are identifiable from its motion [5]. Only certain linear combinations of them called base inertial parameters [8] are. So another important task in analysis of multibody systems is to obtain the identification model in terms of the identifiable parameters and find a connection between them and coefficients of the motion equations. This can be done using the suggested approach.

Consider the energy identification model [5]. The total energy function \( H = K + P \) of a mechanism can be represented using generalized inertial parameters as follows

\[
H = b(q, \dot{q}) \mu.
\]

Here, \( \mu \) contains only those generalized parameters which are nonzero apriori and \( b \) is a row vector which comprises all basis elements of space \( \mathbb{B}_L \) corresponding to these parameters. Taking into account Equation 15 and Equation 16, we obtain

\[
H = b(q, \dot{q}) \mu_{b} = b(q, \dot{q}) C_{\mu}^T C_{h}^T \mu_{b} = b(q, \dot{q}) \mu_{b} = b(q, \dot{q}) C_{h}.
\]

Let us define the row vector \( h(q, \dot{q}) \) of \( r \) influence functions which determine contribution of the basis generalized parameters \( \mu_{b} \) to the total energy \( H \). The influence functions are elements of space \( \mathbb{B}_L \) as they are linear combinations of its basis functions:

\[
h(q, \dot{q}) = b(q, \dot{q}) C_{h},
\]

where \( C_{h} = (I_r C_{\mu}^T) P \). They are also linearly independent since their coordinate matrix \( C_{h} \) has the maximum rank \( r \). Therefore, the basis generalized parameters are identifiable due to the fact that the influence functions generate the regression matrix of the energy model. Generally, the minimal dynamic parameter set and the identifiable parameter set are different concepts. The former spans the subspace of mechanism dynamic parameters and the latter contains parameters which can be uniquely estimated by the motion data. As this section shows, they are the same for serial multibody systems with holonomic constraints but can differ for nonholonomic or closed-loop systems.

We should notice that matrix \( C_{h} \) can be calculated only if geometrical parameters of a mechanism are known. Otherwise, the set of the identifiable parameters consists of the elements of \( \mu \). The latter follows from linear independence of the components of \( b(q, \dot{q}) \), which generate the columns of the regression matrix in that case. However, this approach is almost impossible to implement practically due to the enormous size of the generalized parameter set. A more efficient and accurate way is to perform kinematic identification [9] at first and then use energy model in the form presented above.

### 4 APPLICATION TO SYMBOLICAL COMPUTATIONS

The approach presented above is useful for symbolical generation of motion equations as their coefficients are calculated directly once the set of generalized inertial parameters is given. The
The main advantage of Equation 11 is that the constant quantities are totally separated from the variable ones. So the coefficients $a_{ki}$, $b_{kij}$, and $c_k$ can be computed only once at the initialization stage and then used in the procedure which solves either direct or inverse dynamics. However, Equations 13 and 14 are not suitable for straightforward application since the whole set of tensor indices must be looked through. A more efficient approach is to iterate over a set of all nonzero generalized inertial parameters calculating their contribution to $b_{kij}$ and $c_k$. The pseudocode of such a procedure is presented below.

```
for all $l \in I^n$ do
    for $k = 1..n$ do
        $\eta[k] = \eta_k(l)$
        $\alpha[k] = \alpha_k(l)$
    end for
    if $\mu_0 \neq 0$ then
        for $k = 1..n$ do
            $c_k^{\eta[k]} = \alpha[k] \mu_0$
        end for
    end if
    for $i = 1..n$ do
        for $j = 1..i$ do
            if $\mu_{ij} \neq 0$ then
                for $k = 1..j$ do
                    $b_{ij}^{\eta[k]} = \frac{1}{2} \alpha[k] \mu_{ij}$
                end for
                for $k = j..n$ do
                    $b_{ij}^{\eta[k]} = \frac{1}{2} \alpha[k] \mu_{ij}$
                end for
                if $j \neq i$ then
                    for $k = 1..i$ do
                        $b_{ij}^{\eta[k]} = \frac{1}{2} \alpha[k] \mu_{ij}$
                    end for
                    for $k = i..n$ do
                        $b_{ij}^{\eta[k]} = \frac{1}{2} \alpha[k] \mu_{ij}$
                    end for
                end if
            end if
        end for
    end for
end for
```

An optimized version of this procedure has been developed to avoid redundant operations computing the coefficients $b_{kij}$ which have at least two of three indices $k, i, j$ equal. The next step is to generate code which deals with variables $q$, $\dot{q}$ and $\ddot{q}$. Here, another advantage of Equation 11 can be used. There is a recursive formula for calculation of basis functions of space $\mathbb{B}$:

$$b_k^m = b_{k-1}^m f_k, \quad k = 1, \ldots, n, \quad b_0 = 1,$$

where $m \in I$, $l_k \in I^n$ and $l_k = (l_{k-1}, m)$. Once all nonzero coefficients of motion equations 11 have been determined, a list of corresponding tensor indices $l \in I^n$ is constituted. Iterating over this list and applying this recursive relation, we can generate the main piece of code for both inverse and
direct dynamics. A very similar procedure can be developed for symbolical formulation of the energy identification model.

5 EXAMPLE
In this section, the main concepts of the presented approach are demonstrated for a general two DoF planar robot whose kinematic scheme is shown in Figure 1. We suppose that its links have arbitrary mass distribution, so the inertia matrix $H$ has a general form. According to classical Denavit-Hartenberg notation, we have

$$ k^{-1}T_k = \begin{pmatrix} \cos q_k & -\sin q_k & 0 & a_k \cos q_k \\ \sin q_k & \cos q_k & 0 & a_k \sin q_k \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad k = 1, 2. $$

At first, the coordinate matrix $C$ of generalized parameters in the standard basis of $P$ is calculated. In this simple example, writing down the expressions of kinetic and potential energy has been done manually using Equations 5 and 6. In real applications, this stage should be performed automatically by a computer program. Such software has been already developed and is being tested now. Brief analysis of matrix $C$ has shown that generalized parameters have projections only on ten dimensional subspace of $P$ spanned by the following subset of its standard basis:

$$ p = (I_{1x}^1, I_{1y}^1, S_{1x}^1, S_{1y}^1, S_{1x}^2, S_{1y}^2, m_1, I_{2x}^2, I_{2y}^2, S_{2x}^2, m_2)^T. $$

The columns of $C$ corresponding to other basis elements of $P$ are zero. Then, the basis rows of coordinate matrix have been determined by transforming it to row echelon form. This results in the following matrix $C_1$

$$ C_1 = \begin{pmatrix} 1 & 1 & 2a_1 & 0 & a_1^2 & 1 & 1 & 2a_2 & 0 & a_1^2 + a_2^2 \\ 0 & 0 & 0 & 0 & 1 & 1 & 2a_2 & 0 & a_2^2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2a_1 & 0 & 2a_1 a_2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2a_1 & 0 \\ 0 & 0 & g & 0 & a_1 g & 0 & 0 & 0 & a_1 g \\ 0 & 0 & 0 & -g & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. $$

Figure 1. General two DoF planar robot with revolute joints.
and the following vectors of the basis and dependent generalized parameters
\[
\mathbf{\mu}_b = \begin{pmatrix} \mu_{11}^{(1,1)} & \mu_{21}^{(1,1)} & \mu_{11}^{(1,2)} & \mu_{11}^{(1,3)} & \mu_{0}^{(2,1)} & \mu_{0}^{(3,1)} \end{pmatrix}^T, \\
\mathbf{\mu}_d = \begin{pmatrix} \mu_{22}^{(1,1)} & \mu_{21}^{(1,2)} & \mu_{21}^{(1,3)} & \mu_{0}^{(2,2)} & \mu_{0}^{(3,2)} & \mu_{0}^{(2,3)} \end{pmatrix}^T.
\]

Multiplying \( C_1 \) by the pseudoinverse of \( C_2 \) (not shown here), we obtain the coordinate matrix \( C_\mu \) of dependent parameters:
\[
C_\mu = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1/2 & 0 & 0 \\ 0 & 0 & 0 & g/2a_1 & 0 \\ 0 & 0 & g/2a_1 & 0 & 0 \\ 0 & 0 & 0 & g/2a_1 & 0 \end{pmatrix}.
\]

Taking nonzero elements from \( C_1 \) and \( C_\mu \), we can generate code for computing generalized inertial parameters which contain much fewer redundant operations. It is possible to avoid such operations if the components of these matrices are tested for dependence. However, this is not a trivial issue.

Then, nonzero coefficients \( b^\ell_{ij} \) and \( c^\ell_k \) are calculated. It has been done by a software module which implements the algorithm presented in the previous section. We have the following result:
\[
\begin{align*}
\mu_1^{(1,3)} &= -\frac{1}{2} \mu_{11}^{(1,2)}, & b_{121}^{(1,3)} &= -\frac{1}{2} \mu_{11}^{(1,2)}, & b_{211}^{(1,3)} &= 0 \mu_{11}^{(1,2)}, & b_{122}^{(1,3)} &= -\mu_{21}^{(1,2)}, & b_{121}^{(1,2)} &= \frac{1}{2} \mu_{11}^{(1,3)}, \\
\mu_1^{(2,2)} &= -\frac{1}{2} \mu_{11}^{(1,3)}, & b_{212}^{(2,2)} &= 1 \mu_{11}^{(1,3)}, & b_{122}^{(2,2)} &= -\mu_{21}^{(1,3)}, & c_{1}^{(3,1)} &= -\mu_{0}^{(2,1)}, & c_{1}^{(2,1)} &= \mu_{0}^{(3,1)}, \\
c_{1}^{(3,2)} &= \mu_{0}^{(3,2)}, & c_{2}^{(3,3)} &= -\mu_{0}^{(3,2)}, & c_{1}^{(3,2)} &= -\mu_{0}^{(2,2)}, & c_{2}^{(2,3)} &= -\mu_{0}^{(2,2)}, \\
c_{1}^{(2,3)} &= \mu_{0}^{(3,3)}, & c_{2}^{(3,2)} &= \mu_{0}^{(3,3)}, & c_{1}^{(3,3)} &= -\mu_{0}^{(2,3)}, & c_{2}^{(2,2)} &= \mu_{0}^{(2,3)}. \\
\end{align*}
\]

Nonzero coefficients \( d^\ell_{ij} \) are not presented since they are equal to nonzero generalized parameters \( \mu_{ij} \). The set \( J \) of tensor indices corresponding to nonzero coefficients of the motion equations is equal to \( \{1, 2, 3\} \times \{1, 2, 3\} \). Finally, the motion equations can be represented as follows
\[
\sum_{ij} b_{ij} y^\ell_k = \tau_k, \quad k = 1, 2,
\]
where
\[
\begin{align*}
y_k^{(1,2)} &= \mu_1^{(1,2)} \ddot{q}_1 + \mu_2^{(1,2)} \ddot{q}_2 + \mu_3^{(1,2)} \ddot{q}_2 + \mu_4^{(1,3)} \ddot{q}_3, & y_k^{(1,2)} &= \mu_1^{(1,2)} \ddot{q}_1 - \frac{1}{2} \mu_{11}^{(1,3)} \ddot{q}_1, \\
y_k^{(1,3)} &= \mu_1^{(1,3)} \ddot{q}_1 + \mu_2^{(1,3)} \ddot{q}_2 - \mu_1^{(1,2)} \ddot{q}_1 \ddot{q}_2 - \mu_2^{(1,2)} \ddot{q}_2, & y_k^{(1,3)} &= \mu_1^{(1,3)} \ddot{q}_1 + \mu_{11}^{(1,2)} \ddot{q}_1 + \mu_{11}^{(1,2)} \ddot{q}_1, \\
y_k^{(1,1)} &= \mu_1^{(1,1)} \ddot{q}_1 + \mu_2^{(1,1)} \ddot{q}_2, & y_k^{(1,1)} &= \mu_1^{(1,1)} \ddot{q}_1 + \mu_{11}^{(1,1)} \ddot{q}_1, \\
y_k^{(2,1)} &= \mu_0^{(3,1)} \ddot{q}_1, & y_k^{(2,1)} &= \mu_0^{(3,1)} \ddot{q}_1, \\
y_k^{(1,3)} &= -\mu_0^{(1,1)} \ddot{q}_1, & y_k^{(1,3)} &= \mu_0^{(1,1)} \ddot{q}_1. 
\end{align*}
\]

Basis functions of space \( \mathcal{B} \) are calculated using recursion
\[
\begin{align*}
b_1^{(1)} &= 1, & b_2^{(1,1)} &= 1, & b_2^{(2,1)} &= b_1^{(1)}, & b_2^{(3,1)} &= b_1^{(1)} , \\
b_2^{(2)} &= f_1^{(1)} , & b_2^{(2,2)} &= f_1^{(1)} , & b_2^{(3,2)} &= b_1^{(1)} f_1^{(1)} , \\
b_2^{(3)} &= f_1^{(1)} , & b_2^{(3,3)} &= f_1^{(1)} , & b_2^{(3,3)} &= b_1^{(1)} f_1^{(1)} .
\end{align*}
\]
6 CONCLUSIONS

This paper presents an alternative approach to parameterization of dynamic models of serial multi-body systems. It is based on considering the Lagrangian function as an element of a finite dimensional linear space. Due to its predefined structure, it is possible to find the natural basis of this space. The coordinates of the Lagrangian in this basis, referred to as generalized inertial parameters, determine mechanism dynamics completely. They are equal to a part of constant coefficients of canonical equations up to a predefined multiplier. The rest of these coefficients are simple linear combinations of generalized parameters. A basis subset of these parameters forms the minimal parameter set of mechanism dynamics. This fact determines a straightforward connection between the coefficients of canonical equations and the identifiable parameters. To find the generalized parameter set and, hence, its basis, the linear mapping matrix C must be calculated. A recursive algorithm for its computation has been developed and is being tested at the moment.

The suggested approach allows direct calculation of the coefficients of the motion equations using both ordinary and basis generalized parameters. So it is applicable for symbolical generation of either direct or inverse dynamic model utilizing CAD data as well as experimental estimation data. Theoretically, computational cost of the model formulation might be reduced by avoiding usage of complicated computer algebra methods, in particular, manipulations with trigonometric expressions. This issue is going to be thoroughly investigated in the nearest future by comparing the approach with the existing ones. This has not been done yet since an efficient algorithm for generating the motion equations is still under development. Further plans also include generalization of the presented approach for some types of higher kinematic pairs as well as its adaptation to open kinematic trees.

REFERENCES


Capturing complex nonlinear failure of bolt connections with simple Multibody System models

Detlef H.-J.F. Neuenhaus*, Markus Feldmann#, Urs Joachim Geßler#

*MBLise
Im Hole 39, 44791 Bochum, Germany
dn@mephisto-online.com

# Institute for Steel Structures
RWTH Aachen University
Mies-van-der-Rohe-Straße 1, 52074 Aachen, Germany
ag@stb.rwth-aachen.de

ABSTRACT

An essential element of road safety are the passive safety devices, in particular vehicle restraint systems (VRS). Used initially in the 1960th to reduce the number of accidents with the oncoming traffic, they serve nowadays generally to stop and redirect vehicles, that are errant from the roadway, and to minimize the consequences of accidents. With the increasing importance of the systems from the once simple guardrail systems to more and more complex systems, the type of road and the traffic volume customize new steel structures (see Fig. 1) or concrete structures or a combination of both materials. The suitability of a system is usually ensured through vehicle impact tests, where different types of vehicle classes with given kinematic properties impact against a prototype of the vehicle restraint system.

Figure 1. Modern vehicle restraint system.

Although classification procedures according to the Eurocode standard EN 1317 [1] require physical tests, this is however quite inefficient for the optimization of such complex systems. A more economical form for parameter studies provide simulation calculations, so-called 'virtual tests'. The dynamics of multibody systems has a decisive advantage over the FE methods, as the shape changes are directly mapped to internal force elements. This allows to define the complex vehicle restraint system as highly simplified calculation model, in particular where modelling the bolted joints.

Keywords: bolted joints, ultimate-limit-state, road restrained systems (RRS), nonlinear spring characteristics, multibody system modelling.

1 INTRODUCTION

The whole numerical model for the simulation of a crash test with a vehicle restraint system can be divided into two major subsystems, firstly, the vehicle model and the other is the restraint system. A road restraint system made of steel, simplified exist of steel profiles for the barrier and the posts, which are hinged together by bolts. In these hinges, rotational and longitudinal springs act with hysteretic deformation characteristics to represent the elastic-plastic deformation behavior. The elastic-plastic bending, torsional and longitudinal deformation of the steel sections can be detected by analytical approach, e.g. based on the hinge-beam-model (for
details see [2], [3]), combined with the plastic hinge theory [4]. The longitudinal expansion of the profile itself is negligible since the elongation of the longitudinal elements takes place almost exclusively within the deformations in the bolted connections. Therewith these connections represent the other major substructure in addition to the steel profiles, whose strength and deformation behavior is a dominant component to the interaction of the entire system.

Hereafter at first, an overview of the standard analytical methods in civil engineering for the design of bolted connections is described. Based on this approach, the development is displayed towards a MBS bolted connection model. With the recalculation of an original crash test experiment, the use of such replacement models for bolted connections is presented as a demonstration example.

2 THEORETICAL BACKGROUND

For all bolted connection models, only axial and transverse forces are considered. Bending moments in the bolts and interfaces resulting from eccentric force application are ignored. The neglect of the bending stress is justified since there is no cyclic stress but impact loads are present due to the vehicle crash. The special characteristic of impact loads is that they occur with short-term force peaks, which can significantly exceed the limit loads, but may not lead to failure of the bolt.

These force peaks are filtered over parallel to the longitudinal springs arranged longitudinal damper elements, with a damper constant of 20% of critical damping. As a pattern of many dampers in a connection it can be noted that the attenuation is distributed among all the dampers to the same parts, even if the dampers are unevenly activated.

2.1 Limit state considerations of bolted connections

According to the EC3 [5], the load of a bolted connection is to be determined based on the load capacity of its basic components. This purpose, linear elastic and linear-plastic analysis methods can be used. Depending on the material stiffness the bolted joint is designed due to transverse loads as a result of bearing and shear loads and due to axial loads as a result of tension and punching loads.

Shearing per shear plane (assumption, shank of the bolt is lying in the shear plane):

\[ F_v,Rd = \frac{\alpha_v \cdot f_{ub} \cdot A}{\gamma_{M2}} \]  

Hole bearing:

\[ F_b,Rd = \frac{k_1 \cdot \alpha_b \cdot f_{u} \cdot d \cdot t}{\gamma_{M2}} \]

Tension:

\[ F_t,Rd = \frac{k_2 \cdot f_{ub} \cdot A_s}{\gamma_{M2}} \]

Punching shear:

\[ B_{p,Rd} = \frac{0.6 \cdot \pi \cdot d_m \cdot t_p \cdot f_u}{\gamma_{M2}} \]

where

- \( \alpha_v = 0.6 \) if the shank of the bolt is lying in the shear plane
- $A$ cross section of the shank
- $\alpha_b$ at edge $\min\left\{\frac{e_1}{d_0}; \frac{d_0}{d_0}; 1.0\right\}$, inside $\min\left\{\left(\frac{p_1}{d_0}; \frac{p_2}{d_0}; 1.0\right)\right\}$
- $k_1$ at edge $\min\left\{2.8 \cdot \frac{e_1}{d_0} - 1.7; 1.4 \cdot \frac{e_2}{d_0} - 1.7; 2.5\right\}$, inside $\min\left\{1.4 \cdot \frac{e_2}{d_0} - 1.7; 2.5\right\}$
- $k_2 = 0.9$ coefficient at tension
- $d$ bolt diameter
- $d_0$ hole diameter
- $d_m$ average of width across corners and spanner opening of the bolt head or nut
- $t$ plate thickness
- $t_p$ washer thickness
- $e_1$ edge distance in the direction of force (Figure 2)
- $e_2$ edge distance transverse to the direction of force (Figure 2)
- $p_1$ hole distance in the direction of force (Figure 2)
- $p_2$ hole distance transverse to the direction of force (Figure 2)
- $f_u$ ultimate tensile strength plate
- $f_{u, b}$ ultimate tensile strength of the bolt
- $y_{N_2} = 1.25$ partial safety factor for load capacities bolts

**Figure 2.** Designations of the hole distances.

### 2.2 Capture of the bolted joints in the equations of motion

When you define a replacement model, it comes here is usually not a "perfect picture" of the adjacent construction, since it can be assumed in some areas only idealized, but that the deformation behavior must narrow down the possible "real reactions" from a numerical model.

The elastic-plastic deformation behavior of the connection, respectively the time evolution of the deformation value, has to be carried out depending on the load history via a nonlinear analysis. For this purpose, the static limit determination according to EC3 [5] spring characteristics for the axial load and transverse load are defined based. The spring forces are applied on the deformation strains and represented by means of hysteresis.

The characteristic curve (Figure 3) starts with a linear elastic increase to approximately $0.8 \cdot F_{Rd}$, with the initial stiffness $S_{1,e}$. Up to the ultimate limit force $F_{Rd}$, the characteristic is linear with the stiffness of $S_{2,p}$, then remains constant as a function of hole diameter $d_0$ and plate thicknesses $t$, $t_p$ until to the elongation $\Delta L_{4,p}$ first falls flat on $0.8 \cdot F_{Rd}$ to the strain $\Delta L_{5,p}$, as approximation, and then steeply down to zero, which characterizes the complete failure.

In the equivalent model, the axial and transverse forces are to consider separately (see also Figure 4). Decisive transverse force is the minimum limit load of shearing (Eq. (1)) and hole bearing (Eq. (2)) $\min\{F_{v,Rd}; F_{b,Rd}\}$. For the axial forces the minimum limit load of tension (Eq. (3)) and punching shear (Eq. (4)) $\min\{F_{t,Rd}; B_{p,Rd}\}$. Reductions of the limit load or
interaction considerations result due to simultaneous axial and transverse loading can be neglected.

**Figure 3**: Characteristic spring curve template (schematically).

**Figure 4**: a) Axial loading; b) Transverse loading due to shearing; c) Transverse loading due to bearing.

**Figure 5**: Hole deformations due to punching.

Ideally, the suspension characteristics are determined using measurements from experiments. If there are no experimental results exist, the spring characteristics can be determined taking into account the following aspects.

- Depending on the smaller tensile strength, welded beam or bolt $\min\{f_u; f_{ub}\}$, can be selected for both characteristic curves, axial spring and transverse spring, the same initial stiffness $S_{1,E}$.

- For small plate thicknesses is the deformation behavior of the welded beam decisive.

- The smallest ultimate limit load is decisive. One can assume that the limit force due to shear stress is less than from the axial load on the bolt.

- The material designations indicate the theoretical minimum value of yield strength and tensile strength. The real material values are usually higher. If there are no original material data available, the limit forces are increased by 10%.
- All springs of a connection models are in a direct dependence. If a spring from the connection model exceeds the elongation at break, the spring forces of all other springs of this model are also put to zero.

- Due to lateral load, the plate ruptures in the area of the hole in longitudinal axis direction. In case of axial load, a funnel-shaped deformation image in the hole region is formed. Both types have a lower deformation stiffness than the material alone.

- The hole play can be included in the determination of the strain $\Delta L_{1,\sigma}$. Similarly, the 0.2% elongation at break. The hole play of a slotted hole is recognized separately.

- To avoid errors due to machine accuracy, the stiffness $S_{1,\sigma}$ should not be greater than 10 [MN/m]. If this is not possible, then the spring force is distributed among several parallel guided spring elements.

The parameters for the generation of the spring characteristics can be calculated most easily listed below. Parameters of axial load contain the symbol $\perp$ and the parameters for the shear loads the symbol $\parallel$.

If the two limiting forces for axial and lateral loads were determined, the elastic limit can be calculated as

$$\frac{1}{2} \Delta L_{1,\sigma} = \frac{d_0 - d + R_{\perp, Rd} \cdot \min(e_2; (4 \cdot t + 40))}{d \cdot t \cdot E}; \quad \frac{1}{2} \Delta L_{1,\sigma} = \frac{R_{\parallel, Rd}}{R_{\perp, Rd}} \cdot \Delta L_{1,\sigma}$$  \hspace{1cm} (5)

Since the initial stiffness of both curves is the same, only the calculation due to transverse load is needed

$$S_{1,\sigma} = \frac{0.8 \cdot R_{\perp, Rd}}{\Delta L_{1,\sigma}}$$  \hspace{1cm} (6)

The failure of the bolted connection is made at the fifth characteristic point, the breaking elongation limit. As an approximation, it is calculated

$$\frac{1}{2} \Delta L_{5,p} = d_0; \quad \frac{1}{2} \Delta L_{5,p} = \frac{d}{2} + \frac{d_{\text{cd}}}{10}$$  \hspace{1cm} (7)

The elongation upon reaching the ultimate strength is assumed in the middle between the yield strength and the rupture strength

$$\frac{1}{4} \Delta L_{4,p} = \frac{1}{4} \Delta L_{5,p} - \frac{1}{4} \Delta L_{1,\sigma}; \quad \frac{1}{4} \Delta L_{4,p} = \frac{1}{4} \Delta L_{5,p} - \frac{1}{4} \Delta L_{1,\sigma}$$  \hspace{1cm} (8)

For simplicity, one can use a bilinear characteristic up to the elongation at break, with the stiffness $S_{2,p} = S_{1,\sigma}$. The other extreme case would be if the yielding point were coincide with the ultimate strength. It is $(0.2/0.8 + 1) \cdot \Delta L_{1,\sigma} \leq \Delta L_{2,p} \leq \Delta L_{3,p}$. With the hole play $\Delta d_0 = d_0 - d$ and the limit force ratio factor $\beta = R_{\parallel, Rd} / R_{\perp, Rd}$, the elongation $\Delta L_{2,p}$ can be simplified calculated to

$$\frac{1}{2} \Delta L_{2,p} = \frac{2 \cdot d_0 + 1.5 \cdot \Delta d_0}{8}; \quad \frac{1}{2} \Delta L_{2,p} = \frac{d + 1.5 \cdot \beta \cdot \Delta d_0}{8}$$  \hspace{1cm} (9)

Thus it is clear that the second stiffness is

$$\frac{1}{2} S_{2,p} = \frac{0.2 \cdot R_{\perp, Rd}}{\frac{1}{2} \Delta L_{2,p} - \frac{1}{4} \Delta L_{1,\sigma}}; \quad \frac{1}{2} S_{2,p} = \frac{0.2 \cdot R_{\perp, Rd}}{\frac{1}{2} \Delta L_{2,p} - \frac{1}{4} \Delta L_{1,\sigma}}$$  \hspace{1cm} (10)
and the fourth elongation value can be determined

\[
\begin{align*}
\text{\textsuperscript{11}}\Delta L_{4p} & = \text{\textsuperscript{11}}\Delta L_{5p} - \left( \text{\textsuperscript{11}}\Delta l_{2p} - \text{\textsuperscript{11}}\Delta l_{1e} \right) \\
\text{\textsuperscript{11}}\Delta L_{4p} & = \text{\textsuperscript{11}}\Delta L_{5p} - \left( \text{\textsuperscript{11}}\Delta l_{2p} - \text{\textsuperscript{11}}\Delta l_{1e} \right)
\end{align*}
\]

In the case of the axial load the curve drops sharply, but should amount not be greater than the initial stiffness \( S_{1e} \). Is hole bearing decisive in case of transverse loading, then the strain determination come from the edge distance in force direction.

\[
\text{\textsuperscript{11}}\Delta L_{\text{6c}} = \frac{d_0}{2} + m \min \left\{ e_1 ; (4 \cdot t + 40) \right\} \quad ; \quad \text{\textsuperscript{11}}\Delta L_{\text{6c}} = \text{\textsuperscript{11}}\Delta L_{5p} + \text{\textsuperscript{11}}\Delta L_{1e} \tag{12}
\]

For numerical reasons, a seventh characteristic point \( \Delta L_{\text{7c}} \) may be necessary. This is greater than the elongation \( \Delta L_{\text{6c}} \) and has no spring forces.

To represent a slotted hole, a spring for this translation can be used having a linear elastic characteristic of two levels (see Figure 6). The first level is to describe the friction of the plates during free movement of the bolt in the hole. The associated spring force and strain at the beginning of this level is

\[
F_{1\text{ch}} = 0.1 \cdot R_{1,Rd} \quad ; \quad \Delta s_{1,\text{ch}} = \frac{F_{1\text{ch}}}{R_{1,Rd}} \cdot \text{\textsuperscript{11}}\Delta L_{1e} \tag{13}
\]

With the maximum hole diameter \( d_{0,1} \) the end of this first level is determined by

\[
F_{2\text{ch}} = 1.2 \cdot F_{1\text{ch}} \quad ; \quad \Delta s_{2,\text{ch}} = \frac{d_{0,1} - 1 - \Delta L_{1p}}{2} + \text{\textsuperscript{11}}\Delta L_{1p} \tag{14}
\]

The second level provides infinite resistance. The beginning of level two can be calculated as

\[
F_{3\text{ch}} = 1.5 \cdot R_{1,Rd} \quad ; \quad \Delta s_{3,\text{ch}} = \frac{F_{3\text{ch}} - F_{2\text{ch}}}{R_{1,Rd}} \cdot \text{\textsuperscript{11}}\Delta L_{1e} + \Delta s_{2,\text{ch}} \tag{15}
\]

![Figure 6: Characteristic spring curve template for slotted hole (schematically).](image)

3 FALLTOWER TESTS

To illustrate the potency of the improvements for the internal force elements, they are used in MBS models for simulating vehicle impact scenario on road restraint systems. The simulation results and in addition the results before the modifications are faced with the results from real
full-scale impact tests. It is demonstrated that these relative simple definitions of spring and damper characteristics for capturing complex failure mechanisms in bolted connections leads to distinct better correlation with real failure scenarios.

Figure 7. Trial: height of fall: 3[m]; plate material: S350; plate thickness: 3[mm];
(a) Two pictures of bolt connections with ruptured bolt holes after experiments.
(b) Deflections of the guardrail beam plate. (c) Forces at the lateral supports of the guardrail beam.

4 USE OF THE BOLT CONNECTION MODELS

4.1 Description of the test item example

Figure 8. Test item details.

As a demonstration example for the use of the replacement models of bolted connections is a recalculation of an original crash test at the French vehicle restraint system GS2. A S235JR
steel guardrail system, with a post distance of 2.00[m] (Figure 8a). The posts are of C100x50x5 profiles (Figure 8b), 1500[mm] in length, driven into the asphalt. The guardrail welded beam, A-profile (Figure 8c), of 4315[mm] in length and fixed to welded spacers (Figure 8d).

4.2 Modelling the bolted connection in the MBS

The kinematic movements are separated, so that for each kinetic property an internal force element can act separately (see Figure 10). For the shear between the post and spacer and the guardrail, nonlinear springs with hysteretical characteristic are generated. Linear spring for the vertical slip of the slotted hole (max 50[mm]) between the post and spacer and the horizontal slip of the slotted hole (max.64[mm]) between the guardrail and the spacer are introduced.

The spacer is on both side with a one-bolt-connection fixed. At the posts by means of one M16x40, 5.8 hexagonal head bolt with M16 hexagonal nut. The guardrail beam are fixed to the spacers by means of one M16x40, 5.8 hexagonal head bolt, M16 hexagonal nut and 80x40 rectangular washer.
4.3 Defining the spring characteristics

First, the axial ultimate loads due to tensile and punching, and the transverse ultimate loads due to shear and hole bearing are calculated using the equations (1) to (4), for both M16 bolts of the spacer fixation separately. Due to shear stress is obtained for both bolts the decisive ultimate load for hole bearing to \( R_{L,Rd} = 50.14 \text{ [kN]} \). The decisive axial load type for both bolts is punching, but with different dimensions. At the fixation to post the ultimate force is calculated to \( R_{L,Rd} = 59.96 \text{ [kN]} \). At the fixation to welded beam the ultimate force is much larger \( R_{L,Rd} = 81.45 \text{ [kN]} \).

From this it results, that possibly the spacer failed first at the fixation to post. This is enhanced by the significantly lower elongation at break limit in the spring characteristics of the post-connection.

<table>
<thead>
<tr>
<th>Table 1. Spring characteristics.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Post</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>(post/spacer</td>
</tr>
<tr>
<td>(post/spacer</td>
</tr>
<tr>
<td>(spacer) ( c_1 / c_2 = )</td>
</tr>
<tr>
<td>(spacer) ( e_1 / e_2 = )</td>
</tr>
<tr>
<td>( p_1 / p_2 = )</td>
</tr>
<tr>
<td>min ( {2.5 \cdot c_2 / (d_{01}) \cdot 2.5 } = k_1 = )</td>
</tr>
<tr>
<td>min ( {c_1 / (3 \cdot d_{01}) \cdot f_{ub} / f_u } \cdot 1 = \alpha_b = )</td>
</tr>
<tr>
<td>( k_2 = )</td>
</tr>
<tr>
<td>(spacer</td>
</tr>
<tr>
<td>(post</td>
</tr>
<tr>
<td>(spacer) ( f_u = )</td>
</tr>
<tr>
<td>(post</td>
</tr>
</tbody>
</table>

\[
F_{s,Rd} = n_S \alpha_v \ 1.1 \cdot f_{ub} \cdot A / \gamma_M2 = 77284.99 \quad \text{[N]} \quad \text{Shearing:}
\]

\[
F_{b,Rd} = n_S \ k_1 \ \alpha_b \ 1.1 \cdot f_u \ d / \gamma_M2 = 50140.29 \quad 50140.29 \quad \text{[N]} \quad \text{Hole bearing:}
\]

\[
F_{t,Rd} = n_S \ k_2 \ 1.1 \cdot f_{ub} \ As / \gamma_M2 = 90522.43 \quad 90522.43 \quad \text{[N]} \quad \text{Tension:}
\]

\[
B_{p,Rd} = n_S \ 0.6 \ \pi \ d_m \ t_p \ 1.1 \cdot f_u / \gamma_M2 = 59956.19 \quad 81446.20 \quad \text{[N]} \quad \text{Punching shear:}
\]
4.4 Comparing the results from test and the simulation

The following Figure 12 compares the results obtained in full-scale crash-tests with those of the numerical simulation with regard to the decisive classification values of the deformation \((W_p)\) as well as of vehicle deceleration \((\text{ASI})\). As can be seen, the simulations meet the test results very well and are within the allowable difference, (see column ‘Diff’ and the values in the round brackets).

<table>
<thead>
<tr>
<th>n° support</th>
<th>Post no.</th>
<th>Da (m)</th>
<th>Db (m)</th>
<th>(W_p) (m)</th>
<th>0.34</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>00</td>
<td>00</td>
<td>-</td>
<td>0.34</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>00</td>
<td>00</td>
<td>-</td>
<td>0.44</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>07</td>
<td>05</td>
<td>0.39</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>29</td>
<td>27</td>
<td>0.65</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>57</td>
<td>55</td>
<td>0.90</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>68</td>
<td>69</td>
<td>1.02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>49</td>
<td>49</td>
<td>0.88</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>24</td>
<td>23</td>
<td>0.74</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>06</td>
<td>05</td>
<td>0.40</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>02</td>
<td>02</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>01</td>
<td>01</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>00</td>
<td>00</td>
<td>0.34</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\(|W_p| = (0.6 \ldots 0.8)\) | \(\text{ASI}\) = (20 \ldots 26) |

\[\text{THIV} = 18\]
5 CONCLUSIONS

Although the MBS calculation model shown here represents a strongly reduced simulation model for the simulation of vehicle impact on vehicle restraint systems, a very high accuracy of results is obtained since the specification of the actual material laws is transferred directly onto the internal force elements. This provides a powerful tool for easily manageable calculations of full-scale tests. In particular, the often experience of initially unexplainable behavior in tests can be revealed by these simulation models, especially for instance when dust prevented a clear view of the scenario during the full-scale test.

REFERENCES


Conceptual and numerical aspects of the mixed variational formulation of geometrically exact beam models

Peter Češarek*, Dejan Zupan*, Joachim Linn#

* Faculty of Civil and Geodetic Engineering
  University of Ljubljana
  Jamova 2, 1000 Ljubljana, Slovenia
  [peter.cesarek,dejan.zupan]@fgg.uni-lj.si

# Department Mathematical Methods in Dynamics and Durability
  Fraunhofer Institute for Industrial Mathematics ITWM
  Fraunhofer Platz 1, 67633 Kaiserslautern, Germany
  joachim.linn@itwm.fraunhofer.de

ABSTRACT

In our work, we highlight the conceptual and numerical aspects of mixed type formulations of geometrically exact beams. The governing set of equations obtained from mixed-type variational formulations consist of kinematic equations, constitutive equations, equilibrium equations and their boundary conditions, and a special set of equations denoted here as consistency conditions. The consistency conditions impose the requirement that the cross-sectional stress resultant forces and moments are equal to the constitutive forces and moments, respectively, along each point of the beam centerline. We present three beam formulations with assumed constant strains, which is equivalent to linear interpolation of displacements and helicoidal interpolation of rotations. Two of the formulations employ the consistency conditions as separate equations and are of the mixed-type, while in the third formulation the consistency conditions are eliminated from the governing system of equations. The configuration space of mixed-type formulations is simpler, which improves the performance (faster convergence, larger load or prescribed displacement steps) of the iterative solvers, and results in faster computational times. We demonstrate the performance of the formulations with a numerical example.

Keywords: geometrically exact beam, mixed formulation, configuration-based formulation.

1 INTRODUCTION

We concern ourselves with geometrically exact beam formulations and their numerical applications. A rigorous approach to derive the governing equations of the beam is to apply the principle of virtual work in the static case or its generalization for dynamics – d’Alembert’s principle. An important part of deriving the governing equations is the treatment of the kinematic relations between displacements, rotations and strains. In the classical approach, the strains are taken as dependent variables and are obtained from the kinematic equations by means of interpolation of the configuration variables which are subsequently differentiated. This leads to formulations with displacement and rotational parameters as the primary variables – so called configuration-based formulations. Equilibrium problems can likewise be formulated using the strains as independent variables, where the kinematic equations are added to the virtual work principle with the method of Lagrange multipliers. This leads to mixed formulations where the strains and Lagrange multipliers take the role of primary unknowns of the problem. Examples of such formulations for three-dimensional geometrically exact beam are the strain-based formulation of Zupan and Saje [1] and the intrinsic dynamic formulation presented by Hodges [2].

The history of mixed formulations for beams, shells and solids goes back to the 1960’s, and since then they have been applied to many engineering problems. Their most attractive features are the accuracy of the numerically evaluated internal stresses, they successfully overcome the problems that arise in the elastic analysis of (nearly) incompressible solids, avoid the problem of locking
in shells and beams and are natural for solving inelastic problems [3]. The main disadvantage of mixed formulations is the computational expense that comes with additional unknowns in the formulation. Because of that, the displacement-based formulations remained dominant, and reduced integration approaches were developed to overcome the locking problems. A lot of effort was put into establishing the connection between mixed and reduced/selective integration methods, e.g. [4], which constituted the reduced integration approach as a valid methodology, rather than a computational trick. For geometrically nonlinear elastic planar beams the equivalence between mixed and reduced integration methods has been discussed in [5], where mixed formulations are also claimed to be effective in comparison with displacement-based models.

In our work, we highlight the conceptual aspects of mixed formulations for finite-strain three-dimensional beam theory and compare the numerical efficiency of two formulations of finite elements based on the configuration-based finite elements. A particularly attractive choice of beam elements for flexible multibody applications are elements with assumed constant strains [6], or linear interpolation of displacements and helicoidal interpolation of rotations [7–9]. The strain-based formulation [6] employs configuration variables, strains and generalized forces as the primary unknowns and is of mixed-type (SB-mixed formulation). When the strains are eliminated from the set of primary variables using the analytical integration of the kinematic equations, it leads to a formulation where configuration variables and generalized forces are the only primary variables (CB-mixed formulation). A further reduction of such a formulation involves the extraction of generalized forces from the constitutive equations which are assumed to be satisfied. This leads to a classical configuration-based (CB) formulation where only displacement and rotational parameters are the primary unknowns.

2 BEAM FORMULATIONS

2.1 Beam kinematics and constitutive equations

Geometrically exact rod models [10–13] provide kinematical relations between configuration variables and resultant strain measures of the beam, regardless of the magnitude of displacements and rotations. The adopted strain measures are consistent with the virtual work principle and are defined by the first derivatives of configuration variables with respect to arc length parameter $s$ of the beam centerline. The deformation of the centerline, determined by position vector $\vec{r}$, is given by the translational strain measure $\vec{\gamma}$

$$\vec{r}' = \vec{r} - \vec{\gamma}^0.$$  

(1)

The rate of change of orientation of the moving frame, attached to the beam centerline, with respect to the fixed frame, is given by the rotational strain measure $\vec{\kappa}$. For rotations represented with a rotation matrix $\Lambda$ the corresponding kinematic relation reads

$$\Lambda' = S(\vec{\kappa} - \vec{\kappa}^0) \Lambda,$$  

(2a)

for rotations given in terms of a rotational quaternion $\hat{q}$ it reads

$$\hat{q}' = \frac{1}{2} (\vec{\kappa} - \vec{\kappa}^0) \circ \hat{q}$$  

(2b)

and for rotations parametrized by the rotational vector $\vec{\vartheta}$ ($\Lambda = \Lambda(\vec{\vartheta})$ or $\hat{q} = \hat{q}(\vec{\vartheta})$) the kinematic equation is given by

$$\vec{\vartheta}' = T^{-1}(\vec{\vartheta}) (\vec{\kappa} - \vec{\kappa}^0),$$  

(2c)

where $S$ is the skew-symmetric operator, $T$ is a tangent operator of the rotational map, see e.g. [6], and $\circ$ denotes quaternion multiplication. $\vec{\gamma}^0$ and $\vec{\kappa}^0$ do not depend on the deformation of the beam and represent the initial strains which can be expressed from the known initial configuration of the beam. $\vec{\gamma}^0$ coincides with the normal of the cross section, while $\vec{\kappa}^0$ describes the initial curvature.
or twist of the beam centerline. The solution of the first order ODEs (1) and (2) is determined by initial values
\[ r'(0) = r_n, \]
\[ \Lambda(0) = \Lambda_n, \] (3a)
and must satisfy the boundary conditions at \( s = h \)
\[ r'(h) = r_{n+1}, \]
\[ \Lambda(h) = \Lambda_{n+1}, \] (3b)
where \( n \) and \( n+1 \) denote the boundary nodes and \( h \) is the length of the beam segment: \( s \in [0, h] \).

For the description of the material behaviour we assume a general (nonlinear elastic) form of the constitutive equations
\[ \vec{f}_C = C_F(\vec{\gamma}, \vec{\kappa}), \]
\[ \vec{m}_C = C_M(\vec{\gamma}, \vec{\kappa}), \] (4)
where \( \vec{f}_C \) and \( \vec{m}_C \) are the resultant cross-sectional constitutive force and moment vectors, and the operators \( C_F \) and \( C_M \) are at least once differentiable with respect to \( \vec{\gamma}, \vec{\kappa} \) and \( s \).

2.2 Generalized virtual work principle
Next, we state the equilibrium problem for the beam presented in Figure 1. For simplicity we only consider the static case with a uniformly distributed external load \( \vec{f}_{ext} \) and discrete loads \( \vec{f}_{n,ext}, \vec{f}_{n+1,ext}, \vec{m}_{n,ext}, \vec{m}_{n+1,ext} \) applied at boundaries.

Before we proceed with the variational formulation we must first choose the parametrization of rotations. We could choose either the components of the rotation matrix or quaternion parameters, but such an approach requires that the equilibrium problem is formulated with additional constraints [14, 15]. However, we here choose the representation of rotations with a rotational vector, but we need to stress that the choice of the rotation parameters is not crucial for the comparisons of the formulations presented here.

The virtual work principle [11, 12] for the beam in Figure 1 states
\[
\int_0^h \left( \vec{f}_C \cdot \delta \vec{\gamma} + \vec{m}_C \cdot \delta \vec{\kappa} \right) \, ds = \int_0^h \left( \vec{f}_{ext} \cdot \delta \vec{r} + \vec{m}_{ext} \cdot \delta \vec{\vartheta} \right) \, ds + \vec{f}_{n,ext} \cdot \delta \vec{r}(0) + \vec{m}_{n,ext} \cdot \delta \vec{\vartheta}(0) + \vec{f}_{n+1,ext} \cdot \delta \vec{r}(h) + \vec{m}_{n+1,ext} \cdot \delta \vec{\vartheta}(h). \] (5)

In the mixed approach we treat strains as independent quantities, therefore we enforce the kinematic equations (1) and (2c), which relate the configuration variables and strains, using the method
of Lagrange multipliers. First, the constraining equations are multiplied by Lagrange multipliers $\vec{f}$ and $\vec{m}$ and integrated over the length of the beam

$$\begin{align*}
\int_0^h \vec{f} \cdot (\vec{\gamma} - \vec{\gamma}^0 - \vec{r}') ds &= 0 \quad (6) \\
\int_0^h \vec{m} \cdot (\vec{\kappa} - \vec{\kappa}^0 - T (\vec{\vartheta}) \vec{\vartheta}') ds &= 0.
\end{align*}$$

Secondly, equations (6) are varied with respect to the kinematic variables $\delta \vec{r}$, $\delta \vec{\vartheta}$, $\delta \vec{\gamma}$, $\delta \vec{\kappa}$ and the Lagrange multipliers $\delta \vec{f}$ and $\delta \vec{m}$, and added to virtual work principle (5). This leads to the modified principle of virtual work

$$\begin{align*}
\int_0^h \delta \vec{\gamma} \cdot (\vec{f}_C - \vec{f}) ds + \int_0^h \delta \vec{\kappa} \cdot (\vec{m}_C - \vec{m}) ds \\
- \int_0^h \delta \vec{r} \cdot (\vec{f}' + \vec{f}_{ext}) ds - \int_0^h \delta \vec{\vartheta} \cdot (\vec{m}' + \vec{r}' \times \vec{f}) ds \\
- \int_0^h \delta \vec{f} \cdot (\vec{\gamma} - \vec{\gamma}^0 - \vec{r}') ds - \int_0^h \delta \vec{m} \cdot (\vec{\kappa} - \vec{\kappa}^0 - T (\vec{\vartheta}) \vec{\vartheta}') ds \\
+ \delta \vec{r}(0) \cdot (\vec{f}_{n,ext} + \vec{f}(0)) + \delta \vec{\vartheta}(0) \cdot (\vec{m}_{n,ext} + \vec{m}(0)) \\
+ \delta \vec{r}(h) \cdot (\vec{f}_{n+1,ext} - \vec{f}(h)) + \delta \vec{\vartheta}(h) \cdot (\vec{m}_{n+1,ext} - \vec{m}(h)) &= 0. 
\end{align*}$$

(7)

The coefficients at the independent variations in (7) must vanish, which results in the Euler-Lagrange equations of the three dimensional beam. They consist of:

(i) the kinematic equations

$$\begin{align*}
\vec{\gamma} - \vec{\gamma}^0 - \vec{r}' &= 0 \quad (8) \\
\vec{\kappa} - \vec{\kappa}^0 - T (\vec{\vartheta}) \vec{\vartheta}' &= 0. \quad (9)
\end{align*}$$

(ii) the equilibrium equations

$$\begin{align*}
\vec{f}' + \vec{f}_{ext} &= 0 \quad (10) \\
\vec{m}' + \vec{r}' \times \vec{f} &= 0
\end{align*}$$

and their boundary conditions

$$\begin{align*}
\vec{f}_{n,ext} + \vec{f}(0) &= 0 \quad \vec{f}_{n+1,ext} - \vec{f}(h) &= 0 \\
\vec{m}_{n,ext} + \vec{m}(0) &= 0 \quad \vec{m}_{n+1,ext} - \vec{m}(h) &= 0. 
\end{align*}$$

(11)

(iii) and a special set of equations here denoted as the consistency conditions

$$\begin{align*}
\vec{f}_C - \vec{f} &= 0 \\
\vec{m}_C - \vec{m} &= 0. 
\end{align*}$$

(12)

The equilibrium equations reveal the physical meaning of Lagrange multipliers in our approach, which are found to be the cross-sectional stress resultant force $\vec{f}$ and moment $\vec{m}$ of the beam, while the consistency conditions impose the requirement that $\vec{f}$ and $\vec{m}$ are equal to the constitutive force $\vec{f}_C$ and moment $\vec{m}_C$, respectively, along each point of the beam centerline.
2.3 A general solution approach

Usually it is assumed that the consistency conditions are exactly satisfied, such that they could be eliminated from the governing equations, which reduces the size of the problem. However, after the problem is discretized, the consistency is not necessarily preserved. From this perspective it is advantageous not to eliminate the consistency conditions from the problem, but to keep them as an independent member among the set of the governing equations. In mixed-type finite element formulations the generalized forces are members of the primary variables. This means that the equilibrium-based resultant forces $\vec{f}$ and moments $\vec{m}$, which take the role of Lagrange multipliers associated with the preservation of kinematic constraints, are demanded to be equal to the corresponding constitutive quantities obtained from the strains, using the constitutive equations. This approach completely avoids the shear locking problem without the necessity of any special numerical treatment of the governing equations [1].

Since the consistency equations are not eliminated in our formulation we need to solve the problem for a full set of unknowns: $\vec{r}$, $\vec{\vartheta}$, $\vec{f}$, $\vec{m}$, $\vec{\gamma}$ and $\vec{\kappa}$. The problem is solved by the finite-element method. The unknowns are approximated by discrete values and suitable shape functions. In general we could introduce the interpolation of all the unknowns of the formulation, or – similar to [16] – the interpolation of generalized strains and forces. Here, we follow the approach of Zupan and Saje [1] and select the strains to be the only interpolated variables, while we express the remaining unknowns with strains. By doing so, we minimize the number of unknowns of the mixed formulation. A general solution approach to solve the system of equations (8)–(12) is then as follows:

(i) introduce the interpolation of strain measures,

(ii) integrate the kinematic equations (8)–(9),

(iii) integrate the equilibrium equations (10),

(iv) the remaining equations (consistency conditions (12), static (11) and kinematic (3) boundary conditions) form the governing system of equations, which is solved for the unknown discrete values of interpolated strains, boundary values of the configuration variables, and the discrete-point values of stress-resultants.

The solution approach presented above summarises the conceptual aspect of the mixed (strain-based) formulation presented in [1, 6]. In this computational approach abstract vectors are replaced by their component representations with respect to the global or local – material basis. Strain measures in particular are naturally expressed in material frame. In the following, the vectors expressed in the fixed basis are marked by a lower case font and vectors expressed in the moving base are denoted by an upper case font. The relationship between the two component representations of a vector is established by the rotation matrix as: $u = \Lambda U$.

2.4 Formulations

Mixed strain based formulation

In general any interpolation function can be chosen for the interpolation of strains. Particularly interesting for the multibody applications [9] are the elements with assumed constant material strains

\[ \Gamma(s) = \Gamma, \quad K(s) = K. \]

When the material strains $\Gamma$ and $K$ are taken to be constant, the kinematic equations can be solved analytically. While the kinematic equation for rotations expressed with rotational vector (9) is not appropriate for exact integration, however, the solution of the equivalent equation (2a), written for the material rotational strain, is well known, e.g. [6]; it reads:

\[ \Lambda(s) = \Lambda(0) \exp(s(K-K^0)) = \Lambda(0) \Lambda(s(K-K^0)). \] (13)
Having a closed-form analytical expression for rotations, we can now integrate the kinematic equation (8) to obtain the analytical expression for the position vector

\[ r(s) = r(0) + \int_0^s \Lambda \, (\Gamma - \Gamma^0) \, ds. \]  

(14)

The kinematics of the beam is a boundary value problem, therefore we explicitly demand that the boundary conditions (3) are satisfied: \( r(0) = r_n, \, r(h) = r_{n+1}, \, \Lambda(0) = \Lambda_n, \, \Lambda(h) = \Lambda_{n+1} \). After taking into account (13) and (14) this leads to

\[ r_{n+1} - r_n - \int_0^h \Lambda(s) \, ds \, (\Gamma - \Gamma^0) = 0 \]  

(15)

\[ \Lambda_{n+1} - \Lambda_n \Lambda \left( h \, (K - K^0) \right) = 0. \]  

(16)

Equation (16) is a matrix equation in \( SO(3) \) and consists of nine algebraic equations of which only 3 are independent. Because the exact integration rule for rotations (13) was used, the components of rotation matrix \( \Lambda \left( h \, (K - K^0) \right) \) satisfy the unit length and orthogonality conditions, therefore the remaining six equations are automatically satisfied. The vector representation of equation (16) can be obtained using the extraction of rotational vectors from the rotation matrices. The extraction cannot be expressed by an explicit formula, thus we will use the symbolic notation

\[ [\Lambda_{n+1} - \Lambda_n \Lambda \left( h \, K_{n+1/2} \right)]_{\mathbb{R}^3} = 0. \]

In the present mixed approach we express the stress-resultants with strains. To that end we integrate equilibrium equations (10):

\[ f(s) = f(0) - \int_0^s f_{\text{ext}} \, ds \]

\[ m(s) = m(0) - \int_0^s r' \times f \, ds = m(0) - \int_0^s \Lambda \cdot (\Gamma - \Gamma^0) \times f \, ds. \]  

(17)

The remaining equations – consistency conditions (12), static boundary conditions (11) and kinematic conditions (15) and (16), now constitute the governing system of equations of the present mixed formulation. Because we used the lowest order of interpolation for strains, the integrals over the length of the beam can be evaluated using simple a midpoint rule. The integrated kinematic boundary condition (15) then simplifies to

\[ r_{n+1} - r_n - h \Lambda_{n+1/2} \cdot (\Gamma - \Gamma^0) = 0. \]  

(18)

where \( \Lambda_{n+1/2} = \Lambda_n \Lambda \left( \frac{h}{2} \, (K - K^0) \right) \). After the discretization, the continuous consistency conditions (12) cannot be satisfied at an arbitrary point along the beam centerline. We employ the collocation method and satisfy them at a discrete point. In accord with the integration rule, we choose to satisfy the consistency conditions at the midpoint of the element, i.e. at \( s = s_{n+1/2} = \frac{h}{2} \). We also choose the discrete values of stress-resultants at the midpoint of the element \( f_{n+1/2} = f \left( \frac{h}{2} \right) \) and \( m_{n+1/2} = m \left( \frac{h}{2} \right) \) to be the primary variables and express the boundary values using (17):

\[ f(0) = \frac{h}{2} f_{\text{ext}} + f \left( \frac{h}{2} \right) \]

\[ m(0) = \frac{h}{2} \Lambda_{n+1/2} \left( \Gamma - \Gamma^0 \right) \times \left( f \left( \frac{h}{2} \right) + \frac{h}{2} f_{\text{ext}} \right) + m \left( \frac{h}{2} \right) \]

\[ f(h) = -\frac{h}{2} f_{\text{ext}} + f \left( \frac{h}{2} \right) \]

\[ m(h) = -\frac{h}{2} \Lambda_{n+1/2} \left( \Gamma - \Gamma^0 \right) \times \left( f \left( \frac{h}{2} \right) - \frac{h}{2} f_{\text{ext}} \right) + m \left( \frac{h}{2} \right). \]
The final governing system of equations of the strain-based formulation of the lowest order now reads:

\[
\begin{align*}
g_1 &= f_{n+1/2}^C - f_{n+1/2} = 0, \\
g_2 &= m_{n+1/2}^C - m_{n+1/2} = 0, \\
g_3 &= f_{n,\text{ext}} + \frac{h}{2} f_{\text{ext}} + f_{n+1/2} = 0, \\
g_4 &= m_{n,\text{ext}} + \frac{h}{2} \Lambda_{n+1/2} (\Gamma - \Gamma^0) \times (f_{n+1/2} + \frac{h}{3} f_{\text{ext}}) + m_{n+1/2} = 0, \\
g_5 &= f_{n+1,\text{ext}} + \frac{h}{2} f_{\text{ext}} - f_{n+1/2} = 0, \\
g_6 &= m_{n+1,\text{ext}} + \frac{h}{2} \Lambda_{n+1/2} (\Gamma - \Gamma^0) \times (f_{n+1/2} - \frac{h}{4} f_{\text{ext}}) - m_{n+1/2} = 0, \\
g_7 &= r_{n+1} - r_n - h \Lambda_{n+1/2} (\Gamma_{n+1/2} - \Gamma^0) = 0, \\
g_8 &= [\Lambda_{n+1} - \Lambda_n \Lambda (hK_{n+1/2})]_{\mathbb{R}^3} = 0,
\end{align*}
\]  

(19)

where \( f_{n+1/2}^C = \lambda_{n+1/2} C_F (\Gamma, K) \) and \( m_{n+1/2}^C = \lambda_{n+1/2} C_M (\Gamma, K) \) denote the spatial representations of the constitutive material stress-resultants. The unknowns of system (19) are the kinematic vectors \( r_n, \vartheta_n, r_{n+1}, \vartheta_{n+1} \), the equilibrium stress resultants \( f_{n+1/2}, m_{n+1/2} \) and the strain vectors \( \Gamma_{n+1/2} \) and \( K_{n+1/2} \).

**Mixed configuration based formulation**

Among the primary unknowns of the mixed formulation (19) are the strain vectors. We can eliminate them from the system using equations \( g_7 (18) \) and \( g_8 (16) \)

\[
\begin{align*}
\Gamma_{n+1/2} &= \frac{1}{h} \Lambda_{n+1/2}^T (r_{n+1} - r_n) + \Gamma^0 \\
K_{n+1/2} &= \frac{1}{h} \theta + K^0, \quad \Lambda (\theta) = \Lambda_n \Lambda_{n+1}.
\end{align*}
\]  

(20)

Not surprisingly, the discrete strain approximations (20) are equivalent to the ones obtained by the linear interpolation of displacements and helicoidal interpolation of rotations [17]. After the elimination of the strains from the mixed strain-based formulation (19), we obtain a new partially reduced formulation of the geometrically exact beam, which reads:

\[
\begin{align*}
g_1 &= f_{n+1/2}^C - f_{n+1/2} = 0, \\
g_2 &= m_{n+1/2}^C - m_{n+1/2} = 0, \\
g_3 &= f_{n,\text{ext}} + \frac{h}{2} f_{\text{ext}} + f_{n+1/2} = 0, \\
g_4 &= m_{n,\text{ext}} + \frac{h}{2} \frac{r_{n+1} - r_n}{h} \times (f_{n+1/2} + \frac{h}{3} f_{\text{ext}}) + m_{n+1/2} = 0, \\
g_5 &= f_{n+1,\text{ext}} + \frac{h}{2} f_{\text{ext}} - f_{n+1/2} = 0, \\
g_6 &= m_{n+1,\text{ext}} + \frac{h}{2} \frac{r_{n+1} - r_n}{h} \times (f_{n+1/2} - \frac{h}{4} f_{\text{ext}}) - m_{n+1/2} = 0.
\end{align*}
\]  

(21)

The system (21) represents a mixed-type formulation that needs to be solved for the kinematic vectors \( r_n, \vartheta_n, r_{n+1}, \vartheta_{n+1} \) and the equilibrium stress resultants \( f_{n+1/2} \) and \( m_{n+1/2} \).

**Configuration based formulation**

If we proceed and eliminate the equilibrium stress resultants \( f_{n+1/2} \) and \( m_{n+1/2} \) from equations \( g_1 \) and \( g_2 \) of the system (21) by employing the consistency conditions (12)

\[
\begin{align*}
f_{n+1/2} &= f_{n+1/2}^C, \\
m_{n+1/2} &= m_{n+1/2}^C,
\end{align*}
\]
we obtain the further reduced set of equilibrium equations

\[
\begin{align*}
g_3 &= f_{n,\text{ext}} + \frac{h}{2} f_{\text{ext}} + f_{C,n+1/2}^{\text{C}} = 0, \\
g_4 &= m_{n,\text{ext}} + \frac{h}{2} r_{n+1} - r_n \times \left( f_{n+1/2}^{\text{C}} + \frac{h}{4} f_{\text{ext}} \right) + m_{n+1/2}^{\text{C}} = 0, \\
g_5 &= f_{n+1,\text{ext}} + \frac{h}{2} f_{\text{ext}} - f_{n+1/2}^{\text{C}} = 0, \\
g_6 &= m_{n+1,\text{ext}} + \frac{h}{2} r_{n+1} - r_n \times \left( f_{n+1/2}^{\text{C}} - \frac{h}{4} f_{\text{ext}} \right) - m_{n+1/2}^{\text{C}} = 0,
\end{align*}
\]

(22)
in which only the kinematic vectors \( r_n, \vartheta_n, r_{n+1} \) and \( \vartheta_{n+1} \) are unknown. The formulation (22) can be identified as a classical configuration-based formulation, which is usually obtained using virtual work principle (5) where the strains are expressed with the configuration variables using the discrete kinematic equations (20).

2.5 Notes on numerical implementation

We solve the governing system of equations of the mixed strain-based formulation (19), the mixed configuration-based formulation (21) and the classical configuration-based formulation (22) with Newton’s iterative method. To that end, the systems of equations \( g(x) = 0 \) are linearized as

\[
K \delta x = -g
\]

and then iteratively solved to obtain corrections of the primary variables, until the convergence criteria are satisfied. \( K \) denotes the Jacobian (tangent stiffness) matrix of the system. \( \delta x \) is the vector of unknowns, which consists of the nodal configuration variables \( \delta r_n, \delta \vartheta_n, \delta r_{n+1}, \delta \vartheta_{n+1} \), and in case of mixed formulations, additional internal element variables \( \delta f_{n+1/2}, \delta m_{n+1/2}^{\text{C}} - \text{mixed configuration-based formulation (21)}, \) or \( \delta f_{n+1/2}, \delta m_{n+1/2}^{\text{C}}, \delta r_{n+1/2}, \delta \vartheta_{n+1/2}, \delta k_{n+1/2} - \text{mixed strain-based formulation (19)} \). In practice, the additional variables of the mixed methods are eliminated at the element level by static condensation [18], which can be executed at low computational costs. The efficiency of the static condensation, however, is not a part of this study. The importance of static condensation is twofold, first, it enhances the robustness of the linear solver, and second, by an elimination of the additional variables, the resulting system involves only (primal) configuration degrees of freedom, such that it could be easily incorporated into computational environments based on displacement-based methods.

3 NUMERICAL EXPERIMENT

Bending of \( 45^\circ \) cantilever

In order to demonstrate the performance of the presented formulations (19) – SB-mixed, (21) – CB-mixed and (22) – CB, we analysed a \( 45^\circ \) cantilever bend presented in [19]. The axis of the bending is in the form of the circular arc with the central angle \( 45^\circ \) and radius \( R = 100 \). Bending is located in the horizontal plane \((x,y)\) and subject to an out-of-plane point load \( P = 600 \) in the \( z \) direction at the free-end, see Figure 2, which triggers all modes of deformation of the structure: bending, shear, extension and torsion. The elastic material data and the geometric properties of the cross-section of the beam are: \( E = 10^7 \), \( G = E/2, A_1 = 1, A_2 = A_3 = 5/6, J_1 = 1/6, J_2 = J_3 = 1/12 \). The beam was discretized using 8 equal straight elements.

The first observation from the simulations is that all of the formulations return the same solution. Results for configuration parameters of the tip and internal forces at the midpoint of the fourth element are presented in Figure 2. Next, we compare the execution times for the calculation of element stiffness matrix and the residual vector, and the computational time required by the linear solver in each iteration. Relative values of the time measurements are presented in Figure 3. The linearization of the strain-based formulation is slightly more complex than the linearization of the configuration-based formulations, which results in a larger computational time for the element stiffness matrix (Figure 3(a)). On the other hand, tangent matrices of the mixed formulations
are less dependent on the actual configuration, which results in a faster solution of the linearized system of equations (Figure 3(b)).

- position and orientation of the tip:
  \[ r_x = 15.80 \quad \theta_x = 2.0376 \]
  \[ r_y = 47.23 \quad \theta_y = -0.1390 \]
  \[ r_z = 53.37 \quad \theta_z = 1.5057 \]
- internal forces at midpoint of 4th element:
  \[ F_X = 448 \quad M_X = -2549 \]
  \[ F_Y = 396 \quad M_Y = 1582 \]
  \[ F_Z = 41 \quad M_Z = 12588 \]

**Figure 2.** Bending of 45° cantilever: initial and deformed configuration, and converged results.

![Graph](image)

**Figure 3.** Bending of 45° cantilever: relative computational times for calculation of element tangent matrix and residual, and relative computational times of linear solver.

The latter becomes more important if we observe Newton iterations and total computational times. Results for simulations with different numbers of equal load increments are presented in Table 1.

**Table 1.** Bending of 45° cantilever: convergence of Newton iteration scheme and relative computational times. The iteration stopping criterion was \(10^{-7}\) for the Euclidean norm of the residual.

<table>
<thead>
<tr>
<th>formulation</th>
<th>(n_{inc} = 1)</th>
<th>(n_{inc} = 4)</th>
<th>(n_{inc} = 10)</th>
<th>(n_{inc} = 40)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(n_{iter})</td>
<td>(\sum n_{iter})</td>
<td>(t_{rel})</td>
<td>(n_{iter})</td>
</tr>
<tr>
<td>SB-mixed</td>
<td>7</td>
<td>7</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>CB-mixed</td>
<td>10</td>
<td>10</td>
<td>0.9</td>
<td>8</td>
</tr>
<tr>
<td>CB</td>
<td>/</td>
<td>/</td>
<td>/</td>
<td>13</td>
</tr>
</tbody>
</table>

\(n_{inc}\) – number of load increments, \(n_{iter}\) – typical number of iterations per increment, \(\sum n_{iter}\) – total number of iterations, \(t_{rel}\) – relative computational time

We can observe that the mixed-strain based formulation shows the best convergence properties, however, due to the smaller computational effort, the mixed configuration-based formulation is
faster. The classical configuration based formulation is most dependent on the configuration parameters and typically requires more iterations per load increment compared to the mixed formulations. This can in some cases result in a mediocre performance, as in the present case, where the configuration based formulation does not converge when the load is applied in one step, and is two times slower then the mixed formulations, if the load is applied in four increments. When the solution of the problem is required at a larger number of load stages, e.g. to obtain a smoother response for visualization purposes, all of the formulations have approximately the same computational efficiency.

4 CONCLUSIONS

We compared three geometrically exact beam formulations for large displacement analysis. The governing equations of the formulations were derived from a modified principle of virtual work. Two of the formulations are of mixed-type and employ stress-resultants and strains as independent members of the primary variables. These additional variables add some computational expense, yet they simplify the configuration space, which results in an excellent performance of the mixed formulations. The numerical results proved that all three formulations converge to the same solution and showed that mixed formulations are computationally not only comparably efficient, but can be more efficient as the classical displacement based formulations.

REFERENCES


Geometrically based pseudo-inverse and reciprocal screws of the Jacobian of 4H mechanisms

R. Bartkowiak*, C. Woernle*

* Faculty of Mechanical Engineering and Marine Technology, University of Rostock, Germany 18059 Rostock, {rene.bartkowiak, woernle}@uni-rostock.de

ABSTRACT
An option to analyze overconstrained mechanisms is the local TAYLOR series approximation of the implicitly formulated loop closure conditions. The received equations can be rearranged to eliminate the relative joint coordinates of the mechanism using the pseudo-inverse of the Jacobian matrix and screws which are reciprocal to the column screws of the Jacobian. A geometric method to build up the pseudo-inverse as well as the reciprocal screws is presented for mechanisms with four helical joints (4H).

Keywords: overconstrained, closure condition, jacobian, pseudo-inverse, screw theory.

1 INTRODUCTION
Dual numbers were introduced by CLIFFORD and have the form \( \hat{a} = a + \varepsilon a \varepsilon \) with the real part \( a \), the dual part \( a \varepsilon \) and the dual unit \( \varepsilon \) with \( \varepsilon^2 = 0 \). Vectors can also be extended in dual form. A dual unit vector describes an oriented line in space and is given by

\[
\hat{u} = u + \varepsilon u \varepsilon = u + \varepsilon r \times u, \quad |u| = 1
\]

with the unit vector of direction \( u \) and its moment \( u \varepsilon \) with respect to the reference point \( O_0 \), see Fig. 1a.

A general dual vector describes a screw, comprising an oriented line in space with an associated pitch \( h \) (Fig. 1b),

\[
\hat{a} = a + \varepsilon a \varepsilon = a + \varepsilon (r \times a + ha) = a + \varepsilon (a_{e\perp} + a_{e\parallel}), \quad |a| = a.
\]

It can be decomposed into a dual scalar norm and a dual unit vector,

\[
\hat{a} = a (1 + \varepsilon h)(e + \varepsilon r \times e) = (a + \varepsilon a \varepsilon)(e + \varepsilon e \varepsilon) = \hat{a} \varepsilon.
\]
In the special case $a = 0$ and $a_\epsilon \neq 0$, i.e. $1/h = 0$, the dual representation of a free vector remains, which describes a spatial direction.

The dual notation allows the use of the principle of transference, formulated by Kotelnikov and Study [1]. All calculation rules of ordinary vectors can be transferred to dual vectors, whereby all dual vectors in an equation must have the same reference point. The relations of spherical geometry, described by ordinary vector algebra, can be transferred to the relations of line geometry in space by dual extension of vector equations.

The differential rotation of a vector $\mathbf{a}$ around the rotation axis of $\mathbf{u}$ with the differential angle $d\varphi$, see Fig. 2a, can be described as

$$d\mathbf{a} = \mathbf{u} \times \mathbf{a} \, d\varphi, \quad |\mathbf{u}| = 1. \quad (4)$$

The dual extension of (4),

$$d\hat{\mathbf{a}} = \hat{\mathbf{u}} \times \hat{\mathbf{a}} \, d\hat{\varphi}, \quad \hat{\mathbf{u}} = \mathbf{u} + \epsilon \mathbf{r} \times \mathbf{u}, |\mathbf{u}| = 1, \quad (5)$$

describes the differential displacement of a screw $\hat{\mathbf{a}}$ around the screw axis $\hat{\mathbf{u}}$ with the differential dual angle $d\hat{\varphi} = d\varphi + \epsilon d\ell$, see also Fig. 2b. In case of a pure translation, the pitch is $h \to \infty$ ($d\varphi \to 0$), and expression (5) will be replaced by

$$d\hat{\mathbf{a}} = \epsilon \mathbf{u} \times \mathbf{a} \, dq \quad (6)$$

with the infinitesimal translation $dq \equiv d\ell = h \, d\varphi$ in direction of $\mathbf{u}$.

With the differential equation (5), the derivative of the screw $\hat{\mathbf{a}}$ with respect to the time $t$ is given by

$$\frac{d\hat{\mathbf{a}}}{dt} = \hat{\mathbf{u}} \times \hat{\mathbf{a}} \, \frac{d\hat{\varphi}}{dt} = \hat{\omega} \hat{\mathbf{u}} \times \hat{\mathbf{a}} = \hat{\omega} \times \hat{\mathbf{a}}. \quad (7)$$

Here, $\hat{\omega}$ is the dual angular velocity with the norm $\hat{\omega} = \frac{d\varphi}{dt} + \epsilon \frac{d\ell}{dt} \equiv \omega + \epsilon \nu$, and the dual unit vector $\hat{\mathbf{u}}$ defines the instantaneous screw axis (ISA) of the screw motion of $\hat{\mathbf{a}}$.

For the numerical implementation, it is practical to introduce the motor notation of Von Mises [2]. A motor is a 6-vector

$$\hat{\mathbf{a}} \equiv \begin{bmatrix} \mathbf{a} \\ \mathbf{a}_\epsilon \end{bmatrix} \quad (8)$$

and contains the real and the dual part of a dual vector. In this notation the Klein form of two motors $\hat{\mathbf{a}}$ and $\hat{\mathbf{b}}$ (identity matrix $\mathbf{I}$) is the scalar expression

$$\text{Kl}(\hat{\mathbf{a}}, \hat{\mathbf{b}}) \equiv \hat{\mathbf{a}}^T \Delta \hat{\mathbf{b}} = \mathbf{a}^T \mathbf{b}_\epsilon + \mathbf{a}_\epsilon^T \mathbf{b} \quad \text{with} \quad \Delta = \begin{bmatrix} 0 & \mathbf{I} \\ \mathbf{I} & 0 \end{bmatrix}. \quad (9)$$

Figure 2: Differential displacement. (a) differential rotation. (b) differential screw displacement.
It corresponds to the dual part of the scalar product of two dual vectors. Two screws $\hat{a}, \hat{b}$ are called reciprocal, if the *KLEIN* form of the two motors $\hat{a}, \hat{b}$ vanishes, $\text{Ki}(\hat{a}, \hat{b}) = 0$. The *KILLING* form of two motors is given by

$$\text{Ki}(\hat{a}, \hat{b}) \equiv a^T b. \tag{10}$$

It corresponds to the real part of the scalar product of two dual vectors. The dual vector product in motor notation, also called screw product, is calculated by

$$\hat{a} \times \hat{b} \quad \text{with} \quad \hat{a} \equiv \begin{bmatrix} \tilde{a} & 0 \\tilde{a}_c & \tilde{a} \end{bmatrix}, \tag{11}$$

with the notation $a \times b \equiv \hat{a}$ and the $(6, 6)$ motor dyad $\hat{a}$.

**Loop closure condition at the velocity level of an $nH$-mechanism** In Fig. 3 an open kinematical chain with $n$ helical joints (H-joint) with the screw axes $\hat{a}_i, |a_i| = 1$, and the rotation angles $q_i, i = 1, \ldots, n$, is considered. The screw axis $\hat{a}_1$ of the first joint is fixed to the base.

The contribution of the angular speed $\dot{q}_i$ of the $i$-th helical joint to the spatial velocity of the $n$-th body, comprising the angular velocity $\omega_{n(i)}$ and the velocity $v_{n(i)}$ of point $O$, is given by

$$\begin{bmatrix} \omega_{n(i)} \\ v_{n(i)} \end{bmatrix} = \begin{bmatrix} a_i \\ r_i a_i + h_i a_i \end{bmatrix} \dot{q}_i. \tag{12}$$

The spatial velocity of body $n$ due to the screw motions of all $n$ joints is then given by

$$\begin{bmatrix} \omega_n \\ v_n \end{bmatrix} = \left[ \hat{a}_1 \ldots \hat{a}_n \right] G \begin{bmatrix} \dot{q}_1 \\ \vdots \\ \dot{q}_n \end{bmatrix}, \tag{13}$$

where the $(6, n)$ matrix $G$ is called the geometrical Jacobian.

The kinematical chain is now closed by connecting the $n$-th body with the base, leading to a closed $nH$-mechanism, see Fig. 4.
With the vanishing spatial velocity of the \(n\)-th body, from (13) results the loop closure condition at the velocity level
\[
0 = \begin{bmatrix} \dot{\mathbf{q}}_1 & \cdots & \dot{\mathbf{q}}_n \end{bmatrix} \mathbf{G} = \begin{bmatrix} \dot{\mathbf{a}}_1 \\ \vdots \\ \dot{\mathbf{a}}_n \end{bmatrix}.
\] (14)

### 1.1 NECESSARY MOBILITY CONDITIONS

In the following, overconstrained \(nH\)-mechanisms with \(n \leq 6\) helical joints and \(f = 1\) degree of freedom are considered. With the loop closure condition (14), the sufficient condition for the mobility with \(f = 1\) degree of freedom is
\[
\text{rank} (\mathbf{G}) = n - 1
\] (15)
in an open neighborhood of an actual considered position for which (14) is formulated [3]. The homogeneous equation system (14) can be transformed into the inhomogeneous equation system
\[
\mathbf{A}(s) \lambda(s) = \dot{\mathbf{a}}_n \quad \text{with} \quad \mathbf{A} = \begin{bmatrix} \dot{\mathbf{a}}_1 & \cdots & \dot{\mathbf{a}}_{n-1} \end{bmatrix}, \quad \lambda = \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_{n-1} \end{bmatrix}, \quad \lambda_i = -\frac{\dot{q}_i}{q_n},
\] (16)

The sufficient mobility condition (15) requires that (16) is fulfilled in an open neighborhood of an actual position of the mechanism. Since \(\mathbf{A}\) and \(\lambda\) in (16) are analytical functions of the independent coordinate \(s = q_n\), they can be expanded into Taylor series in the actual position \(s_0 = q_{n0}\), leading to
\[
\dot{\mathbf{a}}_{n0} = \left( A_0 + \frac{A_0'}{1!} \delta s + \frac{A_0''}{2!} \delta s^2 + \cdots \right) \left( \lambda_0 + \frac{\lambda_0'}{1!} \delta s + \frac{\lambda_0''}{2!} \delta s^2 + \cdots \right)
\] (17)

with \(\dot{\mathbf{a}}_{n0} = \dot{\mathbf{a}}_{n,0}, A_0 = A(s_0), A_0' = \frac{\partial A(s)}{\partial s} \big|_{s_0}, \ldots\) and \(\lambda_0 = \lambda(s_0), \lambda_0' = \frac{\partial \lambda(s)}{\partial s} \big|_{s_0}, \ldots\). The derivatives of matrix \(\mathbf{A}_0 = \begin{bmatrix} \dot{\mathbf{a}}_{10} & \cdots & \dot{\mathbf{a}}_{n-1,0} \end{bmatrix}\) with respect to \(s\) are expressed, equivalently to (7), by the derivatives of the screw axes \(\mathbf{a}_k' = \frac{\partial \mathbf{a}_k}{\partial s} \big|_{s_0}\) using the dual vector product,
\[
\dot{\mathbf{a}}_{k0}' = -\sum_{i=1}^{k-1} \mathbf{a}_i \mathbf{a}_k \lambda_i.
\] (18)

As (17) must hold for arbitrary values of \(\delta s\), the coefficients of the powers of \(\delta s\) must independently vanish and by this an infinite number of equivalent algebraical conditions for the mobility
of the mechanism are obtained that are called the closure conditions of order $m = 1, \ldots, \infty$,

$$
m = 1: \quad A_0 \lambda_0 = A_{00} \equiv \hat{\alpha}_1,
$$

$$
m = 2: \quad A_0 \lambda_0' = -A_0^\prime \lambda_0 \equiv \hat{\alpha}_2,
$$

$$
m = 3: \quad A_0 \lambda_0'' = -A_0^\prime \lambda_0 - 2A_0^\prime \lambda_0^\prime \equiv \hat{\alpha}_3.
$$

(19)

The $nH$-mechanisms is mobile if solutions $\lambda_0, \lambda_0', \lambda_0''$ of (19) exist and, consequently, iff every right hand side $\hat{\alpha}_i = \hat{a}_{00}$, $\hat{\alpha}_2 = -A_0^\prime \lambda_0$, $\ldots$ of (19) lies in the column space of matrix $A_0$. As a result all reciprocal screws $\hat{\alpha}_{00}$ obtained from the reciprocity condition

$$
\hat{\alpha}_{00}^T \Delta A_0 = 0, \quad j = 1, \ldots, (6 - \text{rank}(A_0)),
$$

(20)

must also fulfill the reciprocity conditions

$$
\hat{\alpha}_{00}^T \Delta \hat{\alpha}_i = 0, \quad i = 1, \ldots, \infty,
$$

(21)

for every right hand side $\hat{\alpha}_i$ of (19). Then the solution of (19) can be written by means of a left inverse $A_0^+$, thus $A_0^+ A_0 = I$, as

$$
\lambda_0 = A_0^+ \hat{a}_{00},
$$

$$
\lambda_0' = -A_0^+ A_0^\prime \hat{a}_{00},
$$

$$
\lambda_0'' = A_0^+ (2A_0^\prime A_0^0 - A_0^0) A_0^+ \hat{a}_{00}.
$$

(22)

Introducing (22) into (21) yields a system of necessary mobility conditions for the screw axes $\hat{\alpha}_0, i = 1, \ldots, n$, in the actual position of an $nH$-mechanism, as shown in [5],

$$
m = 1: \quad 0 = \hat{\alpha}_{00}^T \Delta \hat{\alpha}_0 \equiv g_1 (\hat{\alpha}_{00}, A_0)
$$

$$
m = 2: \quad 0 = \hat{\alpha}_{00}^T \Delta A_0^\prime A_0^0 \equiv g_2 (\hat{\alpha}_{00}, A_0)
$$

$$
m = 3: \quad 0 = \hat{\alpha}_{00}^T \Delta (-2A_0^\prime A_0^0 + A_0^0) A_0^+ \hat{\alpha}_{00} \equiv g_3 (\hat{\alpha}_{00}, A_0)
$$

$$
\vdots
$$

$$
m: \quad 0 = \ldots \equiv g_m (\hat{\alpha}_{00}, A_0)
$$

(23)

The solution of (23) up to an unknown sufficient finite order $m_{\text{max}}$, depending on the number and type of joints, yields screw coordinates of the joints which guarantee the finite mobility of the mechanism, see also [4]. For several overconstrained mechanisms an estimation of $m_{\text{max}}$ was determined by the numerical solution of (23), refer to [7].

To find an analytical solution of (23) is difficult, as the pseudo-inverse $A_0^+$ as well as the reciprocal screws $\hat{\alpha}_{00}, j = 1, \ldots, (6 - \text{rank}(A_0))$ are not given in an analytical form. In [9] the reciprocal screws of the column screws of $A_0$ are expressed in an analytical form. An analytical form of the pseudo-inverse of $A_0$ using screw theory is presented in [8]. However, these procedures are not practical for an evaluation of (23).

2 GEOMETRICAL PSEUDO-INVERSE AND RECIPROCAL SCREWS

To evaluate the mobility conditions (23), in the following a purely geometrical way to express two special pseudo-inverses as well as a basis of the reciprocal screw space of the Jacobian for 4H mechanisms, thus $n = 4$, is presented. As all terms are evaluated in the actual position of the mechanism, the index 0 is omitted in the following, thus $a_i \equiv a_{0i}$. 

81
Pseudo-inverse The aim is to find a \((3,6)\) pseudo-inverse \(A^+\) of the \((6,3)\) matrix \(A\) with the property \(A^+ A = I\). Since the columns of \(A\) contains the joint screws \(\mathbf{a}_j, j = 1, \ldots, 3\), matrix \(A^+\) can be expressed with three auxiliary screws \(\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3\), for instance in the form

\[
\begin{bmatrix}
\mathbf{b}_1 \\
\mathbf{b}_2 \\
\mathbf{b}_3
\end{bmatrix}^T \Delta \begin{bmatrix}
\mathbf{a}_1 \\
\mathbf{a}_2 \\
\mathbf{a}_3
\end{bmatrix} = I. \tag{24}
\]

Since (24) contains the nine KLEIN forms

\[
\mathbf{b}_i^T \Delta \mathbf{a}_j = \begin{cases} 
1, & \text{for } i = j \\
0, & \text{for } i \neq j
\end{cases}, \quad i, j = 1, \ldots, 3, \tag{25}
\]

each screw \(\mathbf{b}_i\) is reciprocal to the two joint screws \(\mathbf{a}_j, i \neq j\) but not reciprocal to the screw \(\mathbf{a}_j, i = j\).

As known, two screws \(\mathbf{a}, \mathbf{b}\) are reciprocal if their axes intersect orthogonally, see Fig. 5a, thus

\[
\begin{bmatrix}
\mathbf{a} \\
\mathbf{b}
\end{bmatrix}^T \Delta \begin{bmatrix}
\tilde{\mathbf{a}}_a + h_{\mathbf{a}} \mathbf{a} \\
\tilde{\mathbf{b}}_b + h_{\mathbf{b}} \mathbf{b}
\end{bmatrix} = \begin{bmatrix}
\mathbf{a}^T \tilde{\mathbf{b}}_b + \mathbf{b}^T \tilde{\mathbf{a}}_a + (h_{\mathbf{a}} + h_{\mathbf{b}}) \mathbf{a}^T \mathbf{b}
\end{bmatrix} = 0. \tag{26}
\]

The axes of the three screws \(\mathbf{b}_i, i = 1, \ldots, 3\), given by \(\mathbf{b}_1 = \tilde{\mathbf{a}}_a \mathbf{a}_1, \mathbf{b}_2 = \tilde{\mathbf{a}}_a \mathbf{a}_2, \mathbf{b}_3 = \tilde{\mathbf{a}}_a \mathbf{a}_3\), see Fig. 5b, are each common perpendicular lines of the two joint axes \(\mathbf{a}_j, i \neq j\), and with (26) the six conditions of (25) for \(i \neq j\) are fulfilled.

Figure 5: Intersecting screw axes. (a) orthogonally intersecting screw axes. (b) common perpendicular line of two joint axes.

The three conditions of (25) for \(i = j\) yield for each KLEIN form

\[
\mathbf{b}_i^T \Delta \mathbf{a}_j = (\tilde{\mathbf{a}}_a \mathbf{a}_i)^T \Delta \mathbf{a}_3 = \frac{1}{\beta}, \quad i = j = 1, \ldots, 3, \tag{27}
\]

the scaling factor \(\beta\). In this way a special pseudo-inverse of the Jacobian \(A\) is given by

\[
A^+_{\text{geom,1}} = \beta \begin{bmatrix}
\tilde{\mathbf{a}}_a \mathbf{a}_1 \\
\tilde{\mathbf{a}}_a \mathbf{a}_2 \\
\tilde{\mathbf{a}}_a \mathbf{a}_3
\end{bmatrix}^T \Delta, \tag{28}
\]

for which (24) holds.

Additionally a further pseudo-inverse of \(A\) can be determined geometrically. Two screws \(\tilde{\mathbf{a}}, \tilde{\mathbf{b}}\) are reciprocal if screw \(\tilde{\mathbf{b}}\) is a free vector and the axes of the screws are perpendicular, \(\mathbf{a} \perp \mathbf{b}\), thus

\[
\begin{bmatrix}
\mathbf{a} \\
\tilde{\mathbf{r}} \mathbf{a} + h_{\mathbf{a}} \mathbf{a}
\end{bmatrix}^T \Delta \begin{bmatrix}
\mathbf{0} \\
\mathbf{b}
\end{bmatrix} = \mathbf{a}^T \mathbf{b} = 0. \tag{29}
\]
The axes of the three screws \( \hat{\mathbf{b}}_i, i = 1, \ldots, 3 \), given by

\[
\begin{align*}
\hat{\mathbf{b}}_1 &= \begin{bmatrix} 0 \\ \tilde{a}_2 a_3 \end{bmatrix}, & \hat{\mathbf{b}}_2 &= \begin{bmatrix} 0 \\ \tilde{a}_3 a_1 \end{bmatrix}, & \hat{\mathbf{b}}_3 &= \begin{bmatrix} 0 \\ \tilde{a}_1 a_2 \end{bmatrix}
\end{align*}
\]

(30)

are each orthogonally to the axes of the two joint screws \( \hat{\mathbf{a}}_i, i \neq j \), and with (29) the six conditions of (25) for \( i \neq j \) are fulfilled. With the screws (30), the three conditions of (25) for \( i = j \) yield for each KLEIN form

\[
\hat{\mathbf{b}}_i^T \Delta \hat{\mathbf{a}}_j = (\hat{\mathbf{a}}_i a_2)^T a_3 \equiv \frac{1}{\gamma}, \quad i = j = 1, \ldots, 3,
\]

(31)

the scaling factor \( \gamma \). In this way a second special pseudo-inverse of the Jacobian \( A \) is given by

\[
A_{\text{geom},2}^+ = \gamma \begin{bmatrix} 0 & 0 & 0 \\ \tilde{a}_2 a_3 & \tilde{a}_3 a_1 & \tilde{a}_1 a_2 \end{bmatrix}^T \Delta,
\]

(32)

for which (24) holds. The lower part of the pseudo-inverse \( A_{\text{geom},2}^+ \) in (32) is the inverse of the matrix that contains the three unit vectors of the joint screws,

\[
\gamma \begin{bmatrix} \tilde{a}_2 a_3 & \tilde{a}_3 a_1 & \tilde{a}_1 a_2 \end{bmatrix}^T = \left[ \begin{array}{c} a_1 \\ a_2 \\ a_3 \end{array} \right]^{-1},
\]

(33)

as it can be shown with the CRAMER rule.

**Reciprocal screws** For the mobility conditions (23) three screws \( \hat{\mathbf{k}}_1, \hat{\mathbf{k}}_2, \hat{\mathbf{k}}_3 \) are needed that fulfill the reciprocity conditions

\[
\begin{bmatrix} \hat{\mathbf{k}}_1 & \hat{\mathbf{k}}_2 & \hat{\mathbf{k}}_3 \end{bmatrix}^T \Delta \begin{bmatrix} \mathbf{a} \\ \hat{\mathbf{a}}_1 \\ \hat{\mathbf{a}}_2 \\ \hat{\mathbf{a}}_3 \end{bmatrix} = 0,
\]

(34)

that contain the nine KLEIN forms

\[
\hat{\mathbf{k}}_i^T \Delta \hat{\mathbf{a}}_j = 0, \quad i, j = 1, \ldots, 3.
\]

(35)

As shown in (26) the screws \( \hat{\mathbf{k}}_i, i = 1, \ldots, 3 \), whose axes are the common perpendicular lines of the two joint axes \( \hat{\mathbf{a}}_i, i \neq j \), given by

\[
\begin{align*}
\hat{\mathbf{k}}_1 &= \tilde{a}_3 \hat{\mathbf{a}}_3 + \begin{bmatrix} 0 \\ h a_2 a_3 \end{bmatrix}, & \hat{\mathbf{k}}_2 &= \tilde{a}_1 \hat{\mathbf{a}}_1 + \begin{bmatrix} 0 \\ h a_3 a_1 \end{bmatrix}, & \hat{\mathbf{k}}_3 &= \tilde{a}_2 \hat{\mathbf{a}}_2 + \begin{bmatrix} 0 \\ h a_1 a_2 \end{bmatrix},
\end{align*}
\]

(36)

fulfill the six reciprocity conditions of (35) for \( i \neq j \) for an arbitrary pitch \( h \). Introducing the screws (36) into the remaining three conditions of (35) for \( i = j \) yields

\[
\hat{\mathbf{k}}_i^T \Delta \hat{\mathbf{a}}_j = (\hat{\mathbf{a}}_3 a_2)^T a_3 + h (\hat{\mathbf{a}}_1 a_2)^T a_3 = \frac{1}{\beta} + h \frac{1}{\gamma}, \quad i = j = 1, \ldots, 3.
\]

(37)

By rearranging (37) the special pitch

\[
h^* = -\frac{\gamma}{\beta},
\]

(38)

can be found, for which the right hand side of (37) vanishes, fulfilling the reciprocity conditions between \( \hat{\mathbf{k}}_i \) and \( \hat{\mathbf{a}}_j \) for \( i = j = 1, \ldots, 3 \). The three screws

\[
\hat{\mathbf{k}}_{\text{geom}1} = \tilde{a}_3 \hat{\mathbf{a}}_3 + \begin{bmatrix} 0 \\ h a_2 a_3 \end{bmatrix}, \quad \hat{\mathbf{k}}_{\text{geom}2} = \tilde{a}_1 \hat{\mathbf{a}}_1 + \begin{bmatrix} 0 \\ h a_3 a_1 \end{bmatrix}, \quad \hat{\mathbf{k}}_{\text{geom}3} = \tilde{a}_2 \hat{\mathbf{a}}_2 + \begin{bmatrix} 0 \\ h a_1 a_2 \end{bmatrix},
\]

(39)

are each reciprocal to the joint screws \( \hat{\mathbf{a}}_1, \hat{\mathbf{a}}_2, \hat{\mathbf{a}}_3 \). The comparison of the three expressions (28), (32) and (39) yields the relation

\[
\begin{bmatrix} \hat{\mathbf{k}}_{\text{geom}1} & \hat{\mathbf{k}}_{\text{geom}2} & \hat{\mathbf{k}}_{\text{geom}3} \end{bmatrix}^T \Delta = \frac{1}{\beta} (A_{\text{geom},1}^+ - A_{\text{geom},2}^+).
\]

(40)

There exist special positions of 4H mechanisms where the geometrically based expressions \( A_{\text{geom},i}^+ \), \( i = 1, 2 \), from (28), (32) and \( \hat{\mathbf{k}}_{\text{geom}j}, j = 1, \ldots, 3 \), from (39) do not fulfill the conditions (24) and (34). These positions have to be excluded as start positions for the local approximation.
2.1 Application to the BENNETT mechanism

The mobility conditions (23) are evaluated with the geometrically based terms (28) and (39) for the BENNETT mechanism, having \( n = 4 \) revolute joints (4R mechanism), thus the self KLEIN forms are vanishing, \( \alpha_i^T \Delta \omega_i = 2h_i = 0, i = 1, \ldots, 4 \).

1\textsuperscript{st}-order mobility conditions

Introducing (28) and (39) into the first equation of (23) yields

\[
0 = \left[ \hat{k}_{\text{geom}1} \hat{k}_{\text{geom}2} \hat{k}_{\text{geom}3} \right]^T \Delta \hat{a}_4 = \begin{bmatrix} (\hat{a}_2 \hat{a}_3)^T \Delta \hat{a}_4 \\ (\hat{a}_3 \hat{a}_1)^T \Delta \hat{a}_4 \\ (\hat{a}_1 \hat{a}_2)^T \Delta \hat{a}_4 \end{bmatrix} + h^* \begin{bmatrix} (\hat{a}_2 \hat{a}_3)^T a_4 \\ (\hat{a}_3 \hat{a}_1)^T a_4 \\ (\hat{a}_1 \hat{a}_2)^T a_4 \end{bmatrix}. \tag{41}
\]

Completing the solution of the 1\textsuperscript{st}-order closure condition, the first equation of (22) is evaluated,

\[
\lambda = A_{\text{geom}1}^+ \hat{a}_4 = \beta \begin{bmatrix} \tilde{a}_2 \tilde{a}_3 \tilde{a}_3 \tilde{a}_1 \tilde{a}_1 \tilde{a}_2 \end{bmatrix}^T \Delta \hat{a}_4 = \beta \begin{bmatrix} (\hat{a}_2 \hat{a}_3)^T \Delta \hat{a}_4 \\ (\hat{a}_3 \hat{a}_1)^T \Delta \hat{a}_4 \\ (\hat{a}_1 \hat{a}_2)^T \Delta \hat{a}_4 \end{bmatrix}. \tag{42}
\]

Rearranging the 1\textsuperscript{st}-order mobility condition (41) and introducing into (42) yields

\[
\lambda = \gamma \begin{bmatrix} a_2 a_3 \\ a_3 a_1 \\ a_1 a_2 \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & a_3 \end{bmatrix}^{-1} a_4. \tag{43}
\]

2\textsuperscript{nd}-order mobility conditions

Introducing (28) and (39) into the second equation of (23) yields

\[
0 = \left[ \hat{k}_{\text{geom}1} \hat{k}_{\text{geom}2} \hat{k}_{\text{geom}3} \right]^T \Delta A_{\text{geom}1}^+ \hat{A}_4 = \begin{bmatrix} (\hat{a}_2 \hat{a}_1)^T \Delta \hat{a}_4 \\ (\hat{a}_3 \hat{a}_1)^T \Delta \hat{a}_4 \\ (\hat{a}_4 \hat{a}_2)^T \Delta \hat{a}_4 \end{bmatrix} + h^* \begin{bmatrix} (\hat{a}_2 \hat{a}_1)^T a_4 \\ (\hat{a}_3 \hat{a}_1)^T a_4 \\ (\hat{a}_4 \hat{a}_2)^T a_4 \end{bmatrix}. \tag{44}
\]

Rearranging the 1\textsuperscript{st}-order mobility condition (41) and introducing into (44) yields the homogeneous system of equations

\[
0 = \begin{bmatrix} (\hat{a}_2 \hat{a}_1)^T \Delta \hat{a}_2 a_3 \\ (\hat{a}_3 \hat{a}_1)^T \Delta \hat{a}_3 a_2 \\ (\hat{a}_4 \hat{a}_2)^T \Delta \hat{a}_4 a_3 \end{bmatrix} + h^* \begin{bmatrix} (\hat{a}_2 \hat{a}_1)^T a_4 \hat{a}_3 a_2 \\ (\hat{a}_3 \hat{a}_1)^T a_4 \hat{a}_2 a_3 \\ (\hat{a}_4 \hat{a}_2)^T a_4 \hat{a}_3 a_1 \end{bmatrix} = Bb \tag{45}
\]
which has nontrivial solutions $b$ iff $\det(B) = 0$. Completing the solution of the 2nd-order closure condition, the second equation of (22) is evaluated,

$$
\lambda' = \beta^3 \begin{pmatrix}
\tilde{a}_1 \tilde{a}_2 \tilde{a}_3 \\
\tilde{a}_1 \tilde{a}_2 \tilde{a}_3 \\
\tilde{a}_1 \tilde{a}_2 \tilde{a}_3 \\
\tilde{a}_1 \tilde{a}_2 \tilde{a}_3 \\
\tilde{a}_1 \tilde{a}_2 \tilde{a}_3 \\
\tilde{a}_1 \tilde{a}_2 \tilde{a}_3
\end{pmatrix}
= -\beta^3 h^3 \begin{pmatrix}
\tilde{a}_1 \tilde{a}_2 \tilde{a}_3 \\
\tilde{a}_1 \tilde{a}_2 \tilde{a}_3 \\
\tilde{a}_1 \tilde{a}_2 \tilde{a}_3 \\
\tilde{a}_1 \tilde{a}_2 \tilde{a}_3 \\
\tilde{a}_1 \tilde{a}_2 \tilde{a}_3 \\
\tilde{a}_1 \tilde{a}_2 \tilde{a}_3
\end{pmatrix}
$$

(46)

It can be generally shown for all 4H mechanisms that the terms $\lambda, \lambda', \lambda'', \ldots$ and consequently the input-output functions of the mechanisms [6] can be determined solely by the real parts of the equations that describe the spherical part of the motions.

For the solution and the comparison of (45) with the higher-order closure conditions, the Lagrange identity provides terms which solely contain Killing and Klein forms, e.g.

$$
\begin{pmatrix}
\tilde{a}_1 \\
\tilde{a}_2 \\
\tilde{a}_3
\end{pmatrix}
\begin{pmatrix}
\Delta \tilde{a}_1 \\
\Delta \tilde{a}_2 \\
\Delta \tilde{a}_3
\end{pmatrix} + h'(\tilde{a}_1 \tilde{a}_2 \tilde{a}_3)
= \begin{pmatrix}
\tilde{a}_1 \\
\tilde{a}_2 \\
\tilde{a}_3
\end{pmatrix}
\begin{pmatrix}
\Delta \tilde{a}_1 \\
\Delta \tilde{a}_2 \\
\Delta \tilde{a}_3
\end{pmatrix}
= \begin{pmatrix}
\tilde{a}_1 \\
\tilde{a}_2 \\
\tilde{a}_3
\end{pmatrix}
\begin{pmatrix}
\Delta \tilde{a}_1 \\
\Delta \tilde{a}_2 \\
\Delta \tilde{a}_3
\end{pmatrix}
-h'(\tilde{a}_1 \tilde{a}_2 \tilde{a}_3)
$$

(47)

The scalar triple product can be written as the volume of an parallelepiped, expressed by its internal angles between the edges,

$$
\left(\tilde{a}_1 \tilde{a}_2 \tilde{a}_3\right)^2 = 1 + 2 \tilde{a}_1 \tilde{a}_2 \tilde{a}_3 \left(\tilde{a}_1 \tilde{a}_2 \tilde{a}_3 - (\tilde{a}_1 \tilde{a}_2)^2 - (\tilde{a}_1 \tilde{a}_3)^2 - (\tilde{a}_2 \tilde{a}_3)^2\right).
$$

(48)

By the dual extension of (48), using the principle of transference, the expression

$$
h^* = -\frac{\tilde{a}_1 \tilde{a}_2 \tilde{a}_3 \Delta \tilde{a}_3}{(\tilde{a}_1 \tilde{a}_2 \tilde{a}_3)^2}
= -\frac{\tilde{a}_1 \tilde{a}_2 \tilde{a}_3 \Delta \tilde{a}_3}{(\tilde{a}_1 \tilde{a}_2 \tilde{a}_3)^2}
$$

(49)

can be found.

Expressed in Killing and Klein forms the 2nd-order closure condition (45) provides nontrivial solutions iff $\det(B) = 0$, which yields the implicit Bennett condition

$$
\frac{\tilde{a}_1 \tilde{a}_2 \tilde{a}_3 \Delta \tilde{a}_3}{(\tilde{a}_1 \tilde{a}_2 \tilde{a}_3)^2} = \pm \frac{\tilde{a}_1 \tilde{a}_2 \tilde{a}_3 \Delta \tilde{a}_3}{(\tilde{a}_1 \tilde{a}_2 \tilde{a}_3)^2}.
$$

(50)

The solution of (45) then can be evaluated as the vector

$$
b = \begin{pmatrix}
\mu_1 \\
\mu_2 \\
\mu_1 \mu_2
\end{pmatrix}
$$

(51)

which contains the symmetry conditions

$$
\begin{pmatrix}
\tilde{a}_1 \tilde{a}_2 \\
\tilde{a}_2 \tilde{a}_3 \\
\tilde{a}_3 \tilde{a}_4
\end{pmatrix}
\tilde{a}_4 = \pm \tilde{a}_1 \tilde{a}_3 \tilde{a}_3 \tilde{a}_4
\rightarrow \hat{q}_1 = \pm \hat{q}_3,
$$

(52)

Since the conditions (50) and (52) are sufficient for the mobility of the Bennett mechanism [10], the 3rd-order mobility conditions must automatically be fulfilled if (45) holds, what is examined in the following.
**3\textsuperscript{rd}-order mobility conditions** Introducing (28), (39) and (41), (44) into the third equation of (23) and rearranging yields the homogeneous system of equations

\[
0 = \begin{bmatrix}
\bar{k}_{\text{geom}1} & \bar{k}_{\text{geom}2} & \bar{k}_{\text{geom}3}
\end{bmatrix}^T \Delta (-2\mathbf{A}'\mathbf{A}^+_{\text{geom},1}\mathbf{A}' + \mathbf{A}''\mathbf{A}^+_{\text{geom},1}) \bar{a}_i
\]

\[
0 = \begin{bmatrix}
(a_i a_1)^T \Delta \bar{a}_i \bar{a}_1 + (a_i a_2)^T \Delta \bar{a}_i \bar{a}_2 + (a_i a_3)^T \Delta \bar{a}_i \bar{a}_3
\end{bmatrix}^T + h^* \begin{bmatrix}
(a_i a_1)^T \bar{a}_i \bar{a}_1 + (a_i a_2)^T \bar{a}_i \bar{a}_2 + (a_i a_3)^T \bar{a}_i \bar{a}_3
\end{bmatrix}
\]

\[
= \mathbf{B} \mathbf{c} + \mathbf{D} \mathbf{d}
\]

For (53) the LAGRANGE identity provides terms which solely contain KILLING and KLEIN forms, e.g.

\[
(a_i a_1)^T \Delta \bar{a}_i \bar{a}_1 + h^* (a_i a_3)^T \bar{a}_i \bar{a}_3
\]

\[
= \frac{\bar{a}_i^T \Delta \bar{a}_i + \bar{a}_i^T \Delta \bar{a}_i}{\beta} - \frac{\bar{a}_i^T \Delta \bar{a}_i}{\beta}
\]

\[
= - \frac{\bar{a}_i^T \Delta \bar{a}_i}{\beta}
\]

In this way the matrix \( \mathbf{D} \) from (53) simplifies to

\[
\mathbf{D} = -\frac{1}{\beta \nu} \begin{bmatrix}
\bar{a}_i^T \Delta \bar{a}_i & \bar{a}_i^T \Delta \bar{a}_i & 0 & 0 \\
0 & 0 & \bar{a}_i^T \Delta \bar{a}_i & \bar{a}_i^T \Delta \bar{a}_i \\
0 & 0 & 0 & -\bar{a}_i^T \Delta \bar{a}_i
\end{bmatrix}
\]

Expressed in KILLING and KLEIN forms the 3\textsuperscript{rd}-order closure condition (53) for the BENNETT mechanism can be factorized into a matrix \( \mathbf{E} \) and the vector \( \mathbf{b} \) from (45),

\[
\mathbf{E} \mathbf{b} = \mathbf{0}
\]

whereby the columns of \( \mathbf{E} \) are linear combinations of the columns of \( \mathbf{B} \), thus rank([\( \mathbf{B}^T \mathbf{E} \)^T]) = rank(\( \mathbf{B} \)) = 2 if the BENNETT conditions (50) are introduced into \( \mathbf{B} \) and \( \mathbf{E} \). With the vector \( \mathbf{b} \) from (51), as a solution of the 2\textsuperscript{nd}-order closure condition (45), the 3\textsuperscript{rd}-order closure condition in the form (56) holds as well.

3 CONCLUSIONS

For \( n \) mechanisms the pseudo-inverse of the jacobian matrix and the screws which are reciprocal to the column screws of the Jacobian can be expressed by geometrical considerations. As a first step it has been shown for the BENNETT 4R mechanism that the mobility condition as well as the closure condition of order \( m = 3 \) are fulfilled if the mobility conditions of order \( m = 1 \) and 2 are fulfilled. This result can be obtained by introducing recursively the mobility conditions together with the properties of screw products, without explicitly solving the closure conditions.
REFERENCES


Multibody Kinematics. A Topological Formulation Based on Structural-Group Coordinates

Mariano Saura*, Pablo Segado*, Bernardino Muñoz*, Daniel Dopico†, Javier Cuadrado†

* Department of Mechanical Engineering
Universidad Politecnica de Cartagena
Doctor Fleming, s/n, 30202 Cartagena, Spain
saura.sanchez@upct.es; psc1@alu.upct.es

# Mechanical Engineering Laboratory
University of la Coruña
Mendizabal, s/n, 15403 Ferrol, Spain
ddopico@udc.es; javicuad@cdf.udc.es

ABSTRACT

Traditional kinematic formulations that are often used to model and analyse multibody systems have advantages, ones with respect to others, in terms of computational efficiency, which make them more suitable for real-time applications and in terms of versatility, which facilitates the modelling and solving multibody systems. A topological formulation which allows an efficient solution and facilitates the automatic modelling of multibody systems would fill the gap that currently exists between these traditional formulations, taking advantage of their benefits and mitigating their drawbacks. In this paper, a topological formulation based on the decomposition of a multibody system into a set of kinematic chains whose kinematics can be solved independently, based on group-coordinates, is introduced. The main objective is to evaluate the efficiency of this formulation considering different levels of specificity in solving the kinematic chains that defines the kinematic structure of the multibody system. To this end, two scalable systems with up to 550 coordinates have been modelled and solved with a global formulation, as a reference, and with up to four solutions based on group-coordinates formulation. The main conclusions drawn by the analysis carried out in this work shows that this topological method offers a greater modularity, flexibility and efficiency than the global method, so it may be of interest both to develop automatic modelling procedures and improve the efficiency in computation time using this formulation.

Keywords: Kinematic structure, Computational kinematics, Group coordinates.

1 INTRODUCTION

Computational kinematic analysis plays a fundamental role in the study of mechanical systems. It is not only necessary in multibody dynamics formulations, frequently is employed as a first stage in the design of mechanical systems (dimensional and/or kinematic synthesis) and, sometimes, the interest in the multibody system (MBS) is purely kinematic (position analysis, range of movement, transmission angle, etc.). Two different families of formulations are normally used in the kinematic analysis of multibody systems: global and topological.

In the global approach, a simple body joint inspection is enough to identify the degrees of freedom that are constrained by each type of kinematic pair. A set of dependent coordinates (reference point, natural or mixed) is introduced to define the model, e.g. \([x_B \ y_B \ x_C \ y_C \ \theta_1]\) in Fig.1.a. Those dependent coordinates are then related through the corresponding constraint equations due to rigid-body and kinematic-pair conditions.

Topological approaches require a detailed study of the kinematic structure of the multibody system to perform a kinematic analysis. There exist different formulations that exploit the topology of the MBS. In the method based on closed-loop identification, the closed-loops have to be opened so as to yield a tree-like structure of the mechanism (Fig.1.b), and then, the kinematic relations among bodies due to the joints connecting them can be defined, along with the loop-closure equations which relate the system dependent \((\theta_2 \ \theta_3)\) and independent \((\theta_1)\) coordinates [1–3]. In the method based on structural group (SG) decomposition, the multibody system is split into SG (e.g. SG-I and SG-II, Fig.1.c) and the variables and constraint equations that each SG introduces into the
system (SG-I: $\theta_1 x_B y_B$ and SG-II: $x_D y_D$) can be included in kinematic and dynamic formulations to obtain the response of the whole system.

For both global and topological formulations there exist a permanent interest in the scientific community in improving the efficiency of the solution and facilitating the automatic modelling of multibody systems. The global formulation is the best suited for automatic modelling of MBS [4–8] but at expenses of its efficiency. Topological formulations based on the decomposition of the system into independent closed-loops [1–3, 9, 10] are more efficient but they lack generality. The topological approach based on SG decomposition combines the advantages of both formulations and could fill the gap between the two.

Because of its interest, the main objective of this paper is to introduce and evaluate the efficiency of a topological formulation based on the kinematic structure of a MBS. To that end, section 2 resumes the basic concepts of the structural analysis and explains how to obtain the kinematic structure of a MBS. Section 3 introduces the algorithms that solve the kinematics of a MBS using its kinematic structure and section 4 defines two case studies which will be used to study the capabilities and efficiency of the proposed method. Section 5 shows and discusses the results derived from the case studies finally, the main conclusions and future developments of the present work are drawn in sections 6 and 7.

2 KINEMATIC STRUCTURE OF A MULTIBODY SYSTEM

The theory of Structural Analysis defines a Structural Group as any kinematic chain whose number of independent chain inputs $n_c$ coincides with its mobility $L_c$ ($n_c = L_c$). The kinematic chains which satisfy this condition and have neither excessive constrains nor additional DOF due to special geometric considerations among their bodies, are defined as normal SG [11], statically determined SG [12], or desmodromic kinematic chains. Furthermore, kinematic chains that cannot split into SG of smaller number of bodies are denominated simple SG. From the simple SG condition and using the Grübler criterion to determine the mobility of a kinematic chain, a useful expression to check whether a given kinematic chain forms a SG or not is obtained (Eq.1). In Eq.1, $S_c$ indicates the number of degrees of freedom allowed by the $P$ kinematic pairs formed by the $N_m$ mobile bodies.

$$S_c - n_c = 3 \cdot (P - N_m)$$

2.1 Obtaining the MBS Kinematic Structure

The kinematic structure of a multibody system defines which SG it is composed of and the specific order in which their SG kinematics have to be solved. Both graph-analytical and computational methods ( [11] and [13] respectively) can be employed to obtain the kinematic structure of the MBS; due to its simplicity the former one is introduced and applied to a four-bar linkage (Fig.2.a).
In the graph-analytical method, the topology of the MBS is represented by its structural graph (Fig.2.b): vertices correspond to bodies and edges to kinematic pairs. The number of edges connecting two vertices equals the degrees of freedom (DOF) or relative movements allowed between them. Finally, a number of these edges, equal to the $n_c$ independent movements defined between the bodies of the kinematic pair, become bold lines referred to as root edges.

The kinematic structure of the MBS is obtained in a very simple manner as depicted in Fig.2.c-f following four basic steps. **First step:** frame 1 isolation and DOF assignment (Fig.2.c). The DOF allowed by each pair in which the frame participates are assigned to the bodies (2 and 4) that form a kinematic pair with the frame (directed edge). The latter bodies become candidates to be SG.

**Second step:** Search for a SG from shorter to larger length. Each one of the candidates is checked to satisfy Eq.1. Here, body 2 is selected (Fig.2.d). The number $P$ of kinematic pairs in which the bodies of the kinematic chain participate are accounted for as the sum of the internal pairs and, from the external pairs, only those with a directed edge (a DOF has been assigned). Thus, for this solid we find that condition in Eq.1 is satisfied and this body is SG, as: $P = 1$, $S_c = 1$, $n_c = 1$, $N_m = 1$. **Third step:** Re-assign DOF. If a kinematic chain forms a SG, the DOF of its external pairs are assigned to the corresponding external bodies. In the example, body 2 is a SG and assigns the DOF $(2 - 3)$ to the body 3, which now is a new candidate (Fig.2.e). There are no more assignments. **Fourth step:** Turn to Step 2. Bodies 3 and 4 are candidates. Starting from one candidate, e.g. body 3, the parameters of this kinematic chain are: $S_c = 1$; $n_c = 0$; $N_m = 1$; $P = 1$. After substituting in Eq.1, body 3 shows not to be SG. Body 4 has the same parameters than body 3 so it is not a SG either. As it is not possible to form SG with a single body, larger chains have to be considered. Starting from a candidate, e.g. body 3, the chain is expanded by selecting another body that forms a kinematic pair with the candidate. The chain $3 - 4$, whose parameters are: $S_c = 3$; $n_c = 0$; $N_m = 2$; $P = 3$ satisfies equation Eq.1 and therefore is a SG (Fig.2.f).

### 2.2 Structural diagram

The kinematic structure of a mechanism is graphically represented by its structural diagram (Fig.2.g). It is composed by as many circles as SG have been obtained plus one, corresponding to the frame, which is identified with the number 0. The two parameters inside each circle $(N_m, n_c)$ corresponds to the number of movable bodies and input movements of the SG. An arrow joins two circles if any of their bodies forms a kinematic pair, and is directed in the same way that the DOF which have been assigned during the structural analysis, showing the order in which the SG have been obtained and dictating the sequence in which their kinematics have to be solved.

---

**Figure 2:** Four-bar linkage. a) Kinematic graph. b) Structural graph. c) to f) Steps to perform structural analysis through its structural graph. g) Structural diagram.
3 COMPUTATIONAL KINEMATICS BASED ON STRUCTURAL GROUPS

In this section we introduce a method for the kinematic analysis of MBS that takes into consideration its kinematic structure. Two algorithms are described: the sequence defined in a main program to solve the whole MBS, and a generic subroutine needed to solve the kinematics of any SG.

3.1 Overall procedure

The general sequence in the kinematic analysis of a MBS can be followed in Algorithm 1. After a data file that models the MBS has been executed, the main program includes three loops. At each time step (first loop) the simulation time is increased and the values of the independent coordinates of the whole system are defined. Then, for each SG in the kinematic structure (second loop) the SG is identified and, depending on its kind (third loop), the appropriated subroutine is called to solve its kinematics. In Algorithm 2 the aspect of subroutine 3RSG called from the main program is shown; this subroutine solves the kinematics of 3RSG as explained in the next section.

```
Algorithm 1: Kinem: Topol. SG solution
%Read data MBS;
MEdatos;
%MBS Kin. analysis%
for \( t = t_0 : timeStep : t_f \) do
    /* set indep. values */
    \( z = z + \Delta z \)
    for \( ng = 2 : length(MGroups) \) do
        /* solve each SG */
        switch MGroups(ng).kind do
            case MGroups(ng).kind == 1RSG
                CALL Solve_1RSG(*ARGS)
            end
            case MGroups(ng).kind == 3RSG
                CALL Solve_3RSG(*ARGS)
        endsw
    end
end
```

```
Algorithm 2: Kinem. Structural Group solution
Solve_3RSG(*ARGS); /* funct. CALL */
%
% I. Position problem %
error = norm(\( \Phi \))
while error > tolerance do
    evaluate \( \Phi \rightarrow \) Jacob
    extract \( \Phi^d \)
    solve \( q_k^d = q_{k-1}^d - (\Phi^d_{k-1})^{-1} \cdot \Phi_{k-1} \)
    evaluate \( \Phi \rightarrow \) mFi
    error = norm(\( \Phi \))
end

% II. Velocity problem
evaluate \( \Phi_{q} \rightarrow \) Jacob
extract \( \Phi_{q}^d \); extract \( \Phi_{q}^i \)
solve \( \dot{q}^d = - (\Phi_{q}^d)^{-1} \cdot \Phi_{q}^i \)

% III. Acceleration problem
evaluate \( \Phi_{q} \rightarrow \) Fiqqq
evaluate \( - [\Phi_{q}^i \dot{q}^i + \Phi_{q} \ddot{q}] \)
solve \( \ddot{q}^d = - (\Phi_{q}^d)^{-1} \cdot [\Phi_{q}^i \dot{q}^i + \Phi_{q} \ddot{q}] \)
```

3.2 Kinematic analysis of a SG

From the main program, the kinematics of each SG is solved by calling the corresponding subroutine. In order to solve each SG, the appropriate set of group coordinates \( q_{G} \) are selected and the corresponding constraint equations \( \Phi \) are defined. The specific subroutine can be programmed according to the following steps.

**Identify the group coordinates and parameters:** A local coordinate system attached to each body is defined and the appropriate set of coordinates (of any kind) that defines the kinematic chain is selected. We introduce two subsets of group coordinates: dependent \( \varphi \) and independent \( h \). The later might differ from the independent coordinates of the whole system (referred to as \( z \) in many recursive formulations). Other parameters which will depend on the specific SG to solve have to be identified from the geometry of the problem and the results of the computational
Figure 3: a) Scalable four-bar linkage is formed by a crank and one dyad. A number \( k \) of dyads, as many as required, can be added to the former one in order to control the number of coordinates of the model. b) Local coordinate systems and group coordinates in a 3R structural group.

Solve the position problem for the SG: To solve the position problem of any SG, the corresponding constraint equations, in accordance to the selected type of coordinates, have to be defined (Eq.2). For the given set of constraint equations, the terms of the Jacobian matrix \( \Phi_\varphi \) can be analytically or numerically obtained and the Newton-Raphson iterative method can be applied to obtain the values of the dependent group coordinates at each \( k \) iteration step (Eq.2). Depending on the SG geometry, an explicit solution of the position problem might be possible and should be taken into account to reduce the computation time.

\[
\Phi = 0 \quad \rightarrow \quad \varphi_k = \varphi_{k-1} - (\Phi_\varphi)^{-1}_{k-1} \cdot \Phi_{k-1}
\]  

Solve the velocity problem: As the values of the independent group velocities \( \dot{h} \) are known, the velocity problem can be formulated by deriving the constraint equations with respect to time, and solved for the dependent ones (Eq.3). Not only the Jacobian matrix \( \Phi_h \), but the whole expression \( - (\Phi_\varphi)^{-1} \Phi_h \) can be analytically obtained in 2D and 3D structural groups with a reduced number of constraint equations.

\[
\dot{\Phi}(q,t) = 0 \quad \rightarrow \quad \varphi = - (\Phi_\varphi)^{-1} \Phi_h \dot{h}
\]  

Solve the acceleration problem: The acceleration problem for the dependent group coordinates can be solved by deriving the velocity constraint equations with respect to time (Eq.4). Again, if the matrices involved show a reduced dimension, most of the calculations can be analytically performed and included into each SG subroutine so that a solver is not needed.

\[
\Phi_\varphi \ddot{\varphi} + \dot{\Phi}_q \dot{q}_G = 0 \quad \rightarrow \quad \ddot{\varphi} = - (\Phi_\varphi)^{-1} \left[ \Phi_h \ddot{h} + \dot{\Phi}_q \dot{q}_G \right]
\]  

Solve the kinematics of other POIs: Apart from the dependent coordinates, the results of other points of interest (POIs) might be necessary (i.e. centre of mass, or reference points for other SG). The position, velocity and acceleration of a POI that belongs to any body (\( p \in j \), Figure 3.b) is easily obtained by making use of the well known equations of rigid body kinematics.

4 CASE STUDIES

To study the advantages or disadvantages that the topological formulation introduces with respect to a global one, two scalable systems are used: the four-bar linkage and the truck suspension.
Figure 4: Bodies that forms the suspension system in one of the axes of a truck (Left). Structural diagram with SG distribution (Centre). A three axes truck (Right).

4.1 Scalable four-bar linkage
A planar four-bar linkage (Fig. 3.a) consists of a crank $\overline{A_1B_1}$, a rod $\overline{B_1C_1}$ and a rocker $\overline{C_1D_1}$. To make the four-bar linkage scalable, an increasing number $k$ of dyads have to be added. Each dyad introduces two bodies $(2k + 1)$ and $(2k + 2)$ joined with an internal rotation joint $C_k$ and attached to the previous dyad and the frame with two external rotation joints $B_k$ and $D_k$. This method allows us to control the number of constraint equations and coordinates in the model. To carry the kinematic analysis out, the input movement is defined as the crank rotation at a constant velocity $\dot{\theta}_1 = 1 \text{ rad/s}$. All lengths are set equal to 6, except $\overline{A_1B_1} = 2$. In order to let the results be comparable against the second case study, the crank will complete 4.77 turns so that the simulation time lasts for 30 seconds.

4.2 Truck suspension
The second MBS corresponds to the suspension system of a truck axes; it is a scalable MBS as an increasing number of axes can be included in the model. Each one of these axes is a two DOF MBS composed of thirteen bodies and different kind of kinematic joints. Figure 4 shows the different bodies that form each one of the axes (left), its structural diagram (center), and a truck with three axes (right).

The structural diagram shows how many SGs forms a multibody system and the order in which their kinematics has to be solved. As it can be seen in the figure, one axes is formed by five groups: SG-I: $\{2, 3, 4, 5\}$, SG-II: $\{6, 8\}$ which is similar to SG-III: $\{10, 11\}$, and SG-IV: $\{7, 9\}$ which is similar to SG-V: $\{12, 13\}$ (Fig. 5):

- **SG-I.** This is a two DOF structural group with four bodies $S_2 - S_5$ (Fig.5.a). Joints: cardan ($S_2 - S_4$ and $S_2 - S_5$), spherical ($S_1 - S_4$, $S_1 - S_5$ and $S_2 - S_3$), and revolute ($S_1 - S_3$). The independent coordinates are defined as the vertical displacement of both ends of the axes (points $P_8$ y $P_9$). This SG is modelled with 8 points and 9 vectors. Other points of interest $P_4 - P_7$ are needed to solve the kinematics of the remaining part of the SG.

- **SG-II.** This is a SG with null mobility and two bodies $S_2$ y $S_3$ (Fig.5.b). Joints: cardan ($S_2 - S_3$), spherical ($S_1 - S_2$) and revolute ($S_3 - S_4$), being $S_4$ body $S_2$ in SG-I. This SG is modelled with 3 points, 5 vectors and an additional coordinate $\theta$ which defines the rotation of $S_3$ relative to $S_4$.

- **SG-III.** This is a SG with null mobility and two bodies, $S_2$ y $S_3$ (Fig.5.c.). Joints: spherical ($S_1 - S_2$), prismatic ($S_2 - S_3$) and cardan ($S_3 - S_4$), being $S_4$ body $S_2$ in SG-I. This SG is modelled with 2 points, 5 vectors and an additional coordinate $s$ which defines the displacement of $S_2$ relative to $S_3$. 

93
4.3 Methods

A total of six analyses have been carried out in this work to study the main differences between a global formulation and a topological one based on structural groups. These analyses are labelled using the following acronyms: FBL (four-bar linkage), TS (truck suspension), GB (global formulation), TP (topological formulation), and a reference to two sparse solvers: MA27 and MA28.

The purpose of the Global formulation in FBL-GB_MA27 and TS-GB_MA27 is to serve as reference analysis of the four-bar linkage and the truck suspension system, respectively. As many global formulations do, each time the system of equations has to be solved for position, velocity and acceleration analysis, the elements of each vector and matrix involved in those analyses are obtained by calling specific-purpose subroutines; then the whole MBS is modelled and solved. For example, to obtain the rigid body constraint of a body defined by two points in natural coordinates, in a planar MBS, an specific subroutine can be called from the main program. By calling this subroutine, with the corresponding arguments, all rigid-body constraints, in the constraints vector, can be automatically evaluated (i.e. Algorithm 3 and 4).

Algorithm 3: Global: deriving Fi

```
FUNCTION evalRestri cs

%% I. Rigid Body restrics %
for i = 1 : 1 : numRigBdy
    bdy=bdy(i); ptB=pointB(bdy)
    ptC=pointC(bdy)
    CALL restrREdy2P(Fi, ptB, ptC, bdy)
    FiGl c(id(i), 1) = Fi
end
```

Algorithm 4: Evaluate Rigid Body Restrics

```
FUNCTION restrREdy2P(Fi, ptB, ptC, bdy)
L = length(bdy)
Fi = (P2(1) − P1(1))^2 + (P2(2) − P1(2))^2 − L^2
```

FBL-TP analysis: As the structural groups in the four-bar linkage have a reduced number of coordinates \( q_g \), the vector of constraint equations \( \Phi \), the Jacobian matrix \( \Phi_{qG} \), and the vector \( \Phi_{qG} \), have their elements defined with symbolic expressions. Moreover, the inverse of the Jacobian matrix, \( \Phi_{qG}^{-1} \), has also been defined so that a solver is not needed for the kinematic analysis.

TS-TP2_MA27 analysis: for the truck suspension system, the same vectors and matrices as in FBL-TP analysis, with exception of the inverse matrix, has their elements expressed in symbolic form. Solver MA27 (for symmetric semi-definite positive systems) is used to solve the system of equations, so that the products \( \Phi_{qG}^T \times \Phi_{qG} \) and \( \Phi_{qG}^T \times \Phi \) have also been defined symbolically. The analysis TS-TP3_MA28 only differs from TS-TP2_MA27 in the selected solver. Finally, analy-
sis TS-TP1_MA27 differs from TS-TP2_MA27 in that the elements of the vectors and matrices involved in the kinematic analysis are not defined with symbolic expressions, but numerically evaluated as previously shown for the Global formulation. In that sense, the only difference between analyses TS-GB_MA27 and TS-TP2_MA27 is that the former solves the whole system of equations of the complete MBS, and the later solves the MBS by calling each of the SG in the kinematic structure sequentially. This analysis is important as it demonstrates that even though symbolic expressions for vectors and matrices in a SG had not been obtained, it is possible to solve this SG as global formulations would do, while others SG can be solved using a symbolic form.

All the analyses are programmed in FORTRAN, compiled with MS Visual Studio in RelWithDebInfo mode and run on a Intel Core i5-2400 CPU 3.10 GHz, RAM 16 GB, and Windows7 SP1 64 bits. All the simulations run from $t_{ini} = 0$ s to $t_{final} = 30$ s and all the MBS have been modelled with mixed (natural and relative) coordinates.

5 RESULTS AND DISCUSSION

In order to the efficiency of the different solutions be comparable, the position, velocity and acceleration of selected variables from each MBS are compared. As an example, Figure 6.a shows the stroke evolution of the hydraulic element ($SG\{10,11\}$) in the first axes of the truck suspension system with respect to the vertical displacement of point $P_8$ while point $P_9$ remains fixed. Recall that vertical displacement of points $P_8$ and $P_9$ defines the two DOF of each axes in the MBS. Two analyses are represented, TS-GB_MA27 and TS-TP2_MA27, and the results are identical.

Figure 6: (Left) To validate the formulations some results are compared. (Right) CPU time versus number of model coordinates for the kinematic analysis of two case studies: Scalable four-bar linkage (FBL) and truck suspension (TS).

<table>
<thead>
<tr>
<th>NUM. COORD.</th>
<th>FOUR-BAR LINK.</th>
<th>TRUCK SUSPENSION</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>GB MA27</td>
<td>TP</td>
</tr>
<tr>
<td>114</td>
<td>0.045</td>
<td>≪ 0.015</td>
</tr>
<tr>
<td>225</td>
<td>0.123</td>
<td>≪ 0.015</td>
</tr>
<tr>
<td>336</td>
<td>0.264</td>
<td>≪ 0.015</td>
</tr>
<tr>
<td>447</td>
<td>0.421</td>
<td>≪ 0.015</td>
</tr>
<tr>
<td>558</td>
<td>0.607</td>
<td>≪ 0.015</td>
</tr>
</tbody>
</table>

Table 1: CPU time vs number of coordinates for the kinematic analysis of the two case studies under different formulations.
5.1 Calculation time for different approaches

In Figure 6.b and Table 1, the calculation time (CPU time) versus the number of coordinates that defines a MBS is shown for the six kinematic analyses that have been carried out. In all cases, the CPU time evolves linearly as the number of coordinates increases. Dashed lines correspond to the four-bar linkage (FBL): the global sparse solution is represented as FBL-GB_MA27 and the topological solution as FBL-TP. In the later, the CPU time is close to zero even in models up to 550 coordinates (the number of coordinates has to be increased up to 1000 for the system to show a CPU time above 0.015 seconds). The global formulation shows a small CPU time for this mechanism for high number of coordinates.

For the truck suspension system (TS) there are four continuous lines in the graph representing: the global sparse analysis (TS-GB_MA27) and the three topological ones (TS-TP1_MA27, TS-TP2_MA27, TS-TP3_MA28), all of them defined in the previous section. This graph shows that when the vectors and matrices involved in the kinematic analysis are automatically evaluated (TS-GB_MA27 and TS-TP1_MA27) the CPU time is 1.8 times slower than the other topological solutions (TS-TP2_MA27 and TS-TP3_MA28) in which those elements are defined with symbolic expressions. Among the two later topological solutions, solver MA28 seems to be more efficient than MA27 as the number of coordinates increases although the differences are small.

It is interesting to observe that in the global formulation the CPU time is considerable lower in the four-bar linkage than in the suspension system, even though both MBS are defined by the same number of coordinates and make use of the same formulation and solver. When a profile tool is executed to evaluate how the CPU time is distributed among the different subroutines, it can be seen that the main differences among the two solutions are due to the higher sparsity of the Jacobian matrices in the four-bar linkage solution.

5.2 CPU time distribution for different approaches

In order to study more in depth the efficiency of these methods, the Visual profile tool has been used to watch how the CPU time is distributed among all the operations involved in the analysis of the truck suspension system (Fig.7.a): the global formulation TS-GB_MA27 and the topological solutions TS-TP1_MA27 and TS-TP2_MA27.
Together with the total CPU time, this figure shows four main calculations during the solution process: evaluate restrictions $\Phi$ and obtain the Jacobian matrices $\Phi_q$, and solve for the position, velocity and acceleration problem. The position problem consumes most of the calculation time due to the high dimension matrix operations in an Newton-Raphson iterative algorithm with a reduced tolerance ($1 \cdot 10^{-8}$). In the case of TS-TP2_MA27, this represents most of its total CPU time as all the vectors and matrices needed in the analysis (except the inverse of the Jacobian matrix) are defined in the corresponding subroutines.

In all the topological methods the acceleration problem (Eq.4) takes more time than the velocity one (Eq.3), because more vectors have to be evaluated. However, in the global solution the velocity problem consumes more time due to a new factorization of the Jacobian matrix after the position problem is solved, which can be used directly in the acceleration problem.

The time consumption in two topological solutions, TS-TP1_MA27 and TS-TP2_MA27, is shown (Fig.7.b) for each of the following six calculation processes: restrictions, Jacobian and independent terms of the linear systems (identified as $\Phi^\ast$), matrix operations, the three routines of solver MA27, denoted by AD (pivoting), BD (decomposition) and CD (solve), and other internal processes. Pivoting and decomposition are the most time-consuming processes, because of the need to access those routines during the Newton-Raphson algorithm in the position problem. The access to the AD routine in the symbolic solution spends a fewer amount of time than its counterpart, because the pivots obtained in the first iteration can be stored and used in the following ones (the same reason holds for the global solution TS-GB_MA27). But the major difference among these two topological solutions is the time spent in matrix operations which is not necessary in TS-TP2_MA27, as they have been performed in advance.

## 6 CONCLUSIONS

The kinematic structure of a multibody system (MBS), which can be obtained with both graph-analytical and computational methods, decomposes a MBS into kinematic chains called Structural Groups (SG), in a specific order, whose kinematic analysis can be carried out using specific-purpose subroutines. Then, solving the kinematics of each SG in the order stated by its kinematic structure, the kinematic analysis of the whole MBS can be achieved at each time step.

In this work, a topological formulation based on the kinematic structure of a MBS is presented and four different approaches that solve the kinematics of their SGs have been studied. In order to evaluate the advantages of this formulation compared to a global one, two scalable MBS with up to 550 coordinates have been modelled and solved: a planar four-bar linkage with an increasing number of bodies and the truck suspension system with an increasing number of axes.

The scalable four-bar linkage is formed by adding to a rotating crank as many SG of a certain type (called 3R-SG) as desired, to achieve an increasing number of coordinates and constraint equations. As these 3R-SG can be modelled with a set of only six natural coordinates, and only two of them are unknowns, the specific-purpose subroutine that solves the kinematics of these 3R-SG is very efficient; all the matrices, vectors and even the inverse of the Jacobian matrix of the constraint equations of this SG are expressed in symbolic form. This is the first and the most efficient approach to a topological formulation based on the kinematic structure of the MBS, and can be applied to any kind of SG defined with a reduced number of coordinates.

The kinematic structure of the truck suspension system is formed by SG of different complexity with a minimum of fifteen unknowns each, so that the inverse of the Jacobian matrix can not be expressed in symbolic form and a solver is needed; this is, however, the most common situation in spatial MBS. To deal with this situation, two different methods based on the use of specific-purpose subroutines that solve the kinematics of each SG are proposed.

In the first method, all the vectors, matrices and the products among them, that are involved in the kinematic analysis (except the inverse of the Jacobian matrix $\Phi_q^{-1}$) have been defined in symbolic
form. As the topology of each SG is well known, most of these calculations can be performed in advance. Two solutions have been implemented in this symbolic form, TS-TP2_MA27 and TS-TP3_MA28, being the solver (MA27 or MA28, respectively) the only difference.

The second method, and fourth topological approach (TS-TP1_MA27) introduced in this work, uses specific-purpose subroutines to solve the kinematics of each SG, but its level of specificity is considerably reduced with respect to the other three solutions. In this case, the vectors and matrices involved in the kinematic analysis of the SG are numerically evaluated at each time step (not given in a symbolic form), making use of a set of more generic subroutines that would serve to solve any kind of SG. This method is directed towards an automatic modelling and solving of MBS.

The efficiency of the four solutions based on the kinematic structure of the MBS is compared to that of a global formulation, FBL-GB_MA27, that solves the scalable four-bar linkage, and TS-GB_MA27, which solves the truck suspension system. Moreover, a profile tool has been used for a detailed study of the time consumed in all the calculations during the kinematic analysis of the MBS. The results that have been obtained allow us to list some advantages that the topological methods based on the kinematic structure possess with respect to the global methods:

**Flexibility:** The topological approach allows the use of any kind of coordinates $q$ (point reference, relative, natural or mixed coordinates) to model a structural group and solve its kinematics.

**Modularity:** The topological method introduced in this work is a modular approach. The kinematic analysis of each SG can be programmed, optimized and compiled in an independent subroutine which might be included in a extensive library of Structural Groups. This modularity offers several advantages: facilitates the modelling and solving of any MBS, and the analysis is fast and reliable.

**Generality versus Efficiency:** Depending on the number of coordinates needed to model each SG, different methods can be used to solve its kinematics. These methods go from FBL-TP with a maximum efficiency and the higher level of specificity, using symbolic expressions to evaluate the dependent coordinates, to a more generic and less efficient method (TS-TP2_MA27) which can be used for the solution of any kind of SG. The former is devoted to a fast and efficient solution (only available in SG with a reduced number of coordinates) while the latter tends to the automatic modelling of the MBS. Intermediate solutions, TS-TP2_MA27 and TS-TP1_MA28, reduce the CPU time by half with respect to TS-TP2_MA27 and the global approach TS-GB_MA27, independently of the number of coordinates in the MBS. The use of symbolic software seems to be the solution to take profit of the potential advantages of the generic method and obtain specific symbolic solutions.

In addition, as the kinematic analysis of each SG is performed independently from the others, the efficiency of any SG subroutine can be improved by selecting the most appropriated solver depending on the structure of the Jacobian matrix of the system of constraint equations. For example, MA27 if $\Phi_q$ is semi-positive definite, or MA28 if it is not. The LU factorization of the Jacobian matrix of each SG can be stored and used at any time step, which improves efficiency.

7 **FUTURE DEVELOPMENTS**

The topological formulation based on the kinematic structure of the MBS has shown many advantages with respect to global formulation. This fact encourages us to improve the possibilities it brings to the computational kinematic and dynamic analysis of MBS, as well as to solve its drawbacks.

As this is a modular approach, the automatic modelling and solving of a MBS can be improved in two ways: making use of symbolic software to obtain an efficient and optimized solution to the kinematics of any SG, and making use of the kinematic structure of the MBS, obtained with computational methods, to automatically define the analysis sequence.
Apart from improving the level of automation, the efficiency of this method could be also improved by including parallel processing and optimizing the use of the best-suited solver to each SG: conventional linear solvers (LAPACK) or other sparse solvers (PARDISO, WSMP or MUMPS) are only some examples.

Finally, a well known drawback related to all the topological formulations is that the efficiency of the solution, and even the capability to find a solution itself, depends on the kinematic structure of a MBS, which is not unique, and might change during the analysis. Methods to find the more efficient kinematic structure and to allow the solution procedure to change to a different kinematic structure, at any time step, must be developed to make this formulation both efficient and general.

REFERENCES


Nonlinear random vibration of the cable modeled as MDOF system and excited by filtered Gaussian white noise

G. K. Er, V. P. Iu, K. Wang

Stochastic Mechanics Laboratory, Faculty of Science and Technology
University of Macau, Macau SAR, China
gker@umac.mo

Abstract
We investigate the nonlinear random vibration of the cables with small sag and excited by colored noise being filtered Gaussian white noise uniformly distributed on the cable. The nonlinear multi-degree-of-freedom systems are formulated based on the revised equation of motion of the geometrically nonlinear cable with Galerkin method. The nonlinear MDOF systems governing the motion of the cable at a given point are formulated and analyzed when the excitation is the filtered Gaussian white noise governed by linear second-order stochastic differential equation. Numerical results obtained with the state-space-split-EPC method, Monte Carlo simulation, and equivalent linearization method are given and compared. The computational efficiency and numerical accuracy of the three methods are also compared.

Keywords: Cable, Nonlinear, Random vibration, Fokker-Planck-Kolmogorov equation, High dimensionality, State-space-split method.

1 Introduction
The cable and many other systems in science and engineering can be modeled as strongly nonlinear stochastic dynamical (NSD) systems with multiple degrees of freedom (MDOF) and huge number of nonlinear terms. It is known that the analysis on the probabilistic solutions of MDOF-NSD systems or high-dimensional Fokker-Planck-Kolmogorov (FPK) equation has been a challenge for almost a century [1, 2, 3], especially for the high (≥4) dimensional systems with strong nonlinearity or large number of nonlinear terms. There are three methods that were used to analyze the MDOF-NSD systems. The first one is the Monte Carlo simulation (MCS) method that was proposed by Metropolis and Ulam in 1949 in their research about nuclear physics [4, 5, 6]. There are some challenges in using MCS method for analyzing the strongly nonlinear stochastic dynamical systems with multiple degree of freedoms, such as the problems of round-off error, numerical stability, convergence, and requirement for huge number of samples for strongly nonlinear systems. The second one is the equivalent linearization (EQL) method which was proposed by Booton in 1954 in the research about nonlinear random vibration of electronic circuit [7, 8]. It is well known that the EQL method is suitable for analyzing the weakly nonlinear systems excited by Gaussian excitation for obtaining the probabilistic solutions of system responses. The third method named state-space-split and exponential polynomial closure (SSS-EPC) method that was proposed in 2010 for the probabilistic solutions of large MDOF-NSD systems with polynomial type of nonlinearity or solving the Fokker-Planck-Kolmogorov equations in high dimensionality [9, 10]. It was extended for analyzing the systems excited by colored or filtered Gaussian white noise [11]. The SSS method can make the problem of solving the FPK equation in high dimensionality become the problem of solving some FPK equations in low dimensionality or make the large NSD system decoupled into some small NSD systems. Therefore, the FPK equations in low dimensionality can be solved with the exponential polynomial closure method [12, 13]. In this paper, the SSS-EPC method is further used to analyze the probabilistic solutions of the in-plane vibration of the cable with small sag and excited by filtered Gaussian white noise uniformly distributed on the cable. The equation of motion of the cable is a nonlinear partial differential equation in time and space [14, 15]. With Galerkin method, the nonlinear partial differential equation is reduced

100
to MDOF-NSD system. The results obtained with the SSS-EPC method are compared with those obtained with EQL and MCS to show the effectiveness of the SSS-EPC method in this case and the advantage of SSS-EPC method over EQL and MCS in analyzing the formulated MDOF system for the cable with small sag, even and strong nonlinearity, and large number of nonlinear terms.

2 Nonlinear stochastic dynamical system of in-plane cable with small sag

Consider a uniform inclined cable hanging between two supports as shown in Figure 1. The analysis of the nonlinear random vibration of the inclined cable is based on the following assumptions. (1) The flexural rigidity, torsional rigidity, and shear stiffness of the cable are ignored. (2) The components of cable weight and dynamic force parallel to the chord line of cable are so small that they can be neglected compared to the pretension in the cable. The static curve of the cable with initial sag is approximately expressed as parabolic curve. (3) The constitutive relation of cable satisfies the Hooke’s law and the stress on the cross section of cable is uniform.

The initial static sag of the inclined cable is caused by the self-weight of cable, but the sag-to-span ratio is very small due to the high pretension applied in the cable ends in real applications such as the cables in cable-stayed bridge. If the distributed stochastic dynamical force \( f_y(x,t) \) being filtered Gaussian white noise is perpendicular to the chord line of the cable, only the transverse vibration of inclined cable can happen. This vibration is one dimensional. Under the action of uniformly distributed load being filtered Gaussian white noise, the lateral motion of the cable is governed by the following equation of motion [15].

\[
\rho w_{tt} + D(\dot{w}) - H w_{xx} = \frac{EA}{l^2} \left( w_{xx} - \frac{8d}{l^2} \right) \int_0^l \left[ \frac{4d}{l} \left( 1 - \frac{2x}{l} \right) w_x + \frac{1}{2} w_x^2 \right] dx = f_y(x,t) \tag{1}
\]

where \( w \) is the dynamic lateral deflection of the cable, \( E \) is Young’s modulus, \( \rho \) is the mass of the cable per unit length, \( A \) is the area of the cable cross section, \( l \) is the distance between two supports, \( D(\dot{w}) \) denotes damping force, \( d \) is the static sag in the middle span of the cable and it is given by \( d = \frac{\rho g l^2 \cos \theta}{S} \), \( H \) is the horizontal component of the static tensile force in the cable, \( g \) is gravitational acceleration, \( \theta \) is the inclined angle with respect to the horizontal plane, \( L_c = \left[ 1 + \frac{16}{l^2} \left( \frac{d}{l} \right)^2 \right] \) (see [15]), \( f_y(x,t) = q_0 F(t) \), \( q_0 \) is constant, and \( F(t) \) is filtered Gaussian white noise governed by

\[
F(t) + 2\zeta_0 \omega_0 F(t) + \omega_0^2 F(t) = \alpha W(t) \tag{2}
\]

where \( \alpha, \zeta_0, \) and \( \omega_0 \) are constants; \( W(t) \) is Gaussian white noise with zero mean and autocorrelation \( E[W(t)W(t+\tau)] = S\delta(\tau) \) in which \( \delta(\tau) \) is Dirac’s delta function and \( S \) is a constant representing the power spectral density (PSD) of \( W(t) \).
3 Multi-degree-of-freedom nonlinear stochastic dynamical system of the cable

If only the linear vibration of the cable is considered, the mode functions of the cable with boundary conditions \( w(0) = w(l) = 0 \) are given by

\[
\phi_i(x) = \frac{1 - \tan(\alpha/2) \sin(\alpha x/l) - \cos(\alpha x/l)}{1 - \tan(\alpha/2) \sin(\alpha/2) - \cos(\alpha/2)}
\]

(3)

When the ratio of sag to span is small, the mode function of the cable with boundary conditions \( w(0) = w(l) = 0 \) can be approximately expressed by \( \phi_i(x) = \sin^{2} \frac{x}{l} \).

The deflection of the cable is expressed with the mode functions by

\[
w(x, t) = \sum_{i=1}^{n} q_i(t) \phi_i(x)
\]

(4)

With Galerkin’s method and the linear mode functions being used as shape functions, the following nonlinear stochastic dynamical system is formulated with Equation (1) if the viscous damping for each mode is assumed to be the same with damping ration \( \xi \).

\[
\ddot{q}_m(t) + 2\xi \omega_m \dot{q}_m(t) + \omega_m^2 q_m(t) + \sum_{i=1}^{n} \sum_{j=1}^{n} b_{ijm} q_i(t) q_j(t) + \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} c_{ijkm} q_i(t) q_j(t) q_k(t) = g_m F(t)
\]

\[(m = 1, 2, \cdots, n)\]

(5)

where \( \omega_m \) is the \( m \)th natural frequency of the linear cable, and

\[
g_m = \frac{q_0}{a_m} \int_0^l \phi_m(x)(x) \, dx
\]

(6)

\[
a_m = \rho \int_0^l \phi_m(x) \, dx
\]

(7)

\[
b_{ijm} = \frac{1}{a_m} \int_0^l B_{ij}(x) \phi_m(x) \, dx
\]

(8)

\[
B_{ij}(x) = \frac{4EA j \pi^2 d}{Lc d^4} \left( i \int_0^l \cos \frac{i \pi x}{l} \cos \frac{j \pi x}{l} \, dx + 2 j \sin \frac{j \pi x}{l} \int_0^l \sin \frac{i \pi x}{l} \, dx \right)
\]

(9)

\[
c_{ijkm} = \frac{1}{a_m} \int_0^l C_{ijk}(x) \phi_m(x) \, dx
\]

(10)

\[
C_{ijk}(x) = \frac{EA j k \pi^2 d^4}{2Lc d^4} \sin \frac{k \pi x}{l} \sin \frac{m \pi x}{l} \int_0^l \cos \frac{i \pi x}{l} \cos \frac{j \pi x}{l} \, dx
\]

(11)

If the joint PDF of the deflection \( w_0(t) = \sum_{i=1}^{n} q_i(t) \phi_i(x_0) \) and velocity \( \dot{w}_0(t) = \sum_{i=1}^{n} \dot{q}_i(t) \phi_i(x_0) \) at \( x = x_0 \) is interested, the following system can be formulated with Equations (4) and (5).

\[
\dot{w}_0(t) + 2\xi \sum_{m=1}^{n} \phi_m(x_0) \omega_m q_m(t) + \sum_{m=1}^{n} \phi_m(x_0) \omega_m^2 q_m(t) + \sum_{m=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} \phi_m(x_0) b_{ijm} q_j(t) q_k(t)
\]

\[+ \sum_{m=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} \phi_m(x_0) c_{ijkm} q_i(t) q_j(t) q_k(t) = F(t) \sum_{m=1}^{n} \phi_m(x_0) g_m
\]

\[
\ddot{q}_m(t) + 2\xi \omega_m \dot{q}_m(t) + \omega_m^2 q_m(t) + \sum_{j=1}^{n} b_{ijm} q_i(t) q_j(t) + \sum_{j=1}^{n} \sum_{k=1}^{n} c_{ijkm} q_i(t) q_j(t) q_k(t) = g_m F(t)
\]

\[(m = 2, 3, \cdots, n)\]

(13)

Equations (12)-(13), Equation (2) and \( q_1(t) = \phi_1^{-1}(x_0)[w_0(t) - \sum_{i=2}^{n} q_i(t) \phi_i(x_0)] \) formulate a NSD system with \( n + 1 \) degrees of freedom about \( w_0, q_2, q_3, \cdots, q_n, \) and \( F(t) \).
4 Dimensionality reduction with state-space-split method

In the following discussion, the summation convention applies unless stated otherwise. The random state variable or vector is denoted with capital letter and the corresponding deterministic state variable or vector is denoted with the same letter in lowercase.

The system governed by Equations (12) and (13) can be generally expressed by the following MDOF-NSD systems.

\[ \dot{A}_i + c_{ij} \dot{A}_j + h_{ii}(A) = h_i F(t) \quad i, j = 1, 2, \ldots, n_A \]  

(14)

where \( A_i \in \mathbb{R}, (i = 1, 2, \ldots, n_A) \), are the components of the vector process \( A \in \mathbb{R}^{n_A} \); \( h_{ii}(A) \) are the polynomial type of nonlinear functions of \( A \) and \( h_{ii}(A) : \mathbb{R}^{n_A} \rightarrow \mathbb{R} \); \( h_i \) and \( c_{ij} \) are constants; \( F(t) \) is the excitation which is assumed to be the filtered Gaussian white noise governed by Equation (2).

Setting \( A_i = X_{2i-1}, \dot{A}_i = X_{2i}, f_{2i-1} = X_{2i}, f_{2i} = h_i F - c_{ij} \dot{A}_j - h_{ii}(A) \), \( g_{2i-1} = g_{2i} = 0 \) \((i = 1, 2, \ldots, n_A)\), \( F = X_{2n_A+1}, \dot{F} = X_{2n_A+2}, f_{2n_A+1} = X_{2n_A+2}, f_{2n_A+2} = -2 \xi_0 \omega_0 X_{2n_A+2} - \alpha_0^2 \omega_0^2 X_{2n_A+1}, g_{2n_A+1} = 0, g_{2n_A+2} = \alpha \), and \( n_k = 2(n_A + 1) \), then Equation (14) and Equation (2) can be jointly expressed by the following coupled Langevin equations or Ito differential equations.

\[ \frac{d}{dt} X_i = f_i(X) + g_i W(t) \quad i = 1, 2, \ldots, n_k \]  

(15)

where \( X \in \mathbb{R}^{n_k}; X_i \ (i = 1, 2, \ldots, n_k) \), are the components of the state vector process \( X; f_i(X) \) : \( \mathbb{R}^{n_k} \rightarrow \mathbb{R} \).

The state vector process \( X \) is Markovian and the probability density function (PDF) \( p(x, t) \) of the Markovian vector process is governed by FPK equation. The stationary PDF \( p(x) \) of the Markovian vector is governed by the following reduced FPK equation [16].

\[ \frac{\partial}{\partial x_i} [f_i(x)p(x)] - \frac{\alpha^2 S}{2} \frac{\partial^2}{\partial x_i^2} p(x) = 0 \]  

(16)

where \( x \) is the deterministic state vector, \( x \in \mathbb{R}^{n_k} \).

It is assumed that the solution of Equation (16) fulfills the following conditions:

\[ \lim_{x_i \rightarrow \pm \infty} f_i(x)p(x) = 0 \quad i = 1, 2, \ldots, n_k \]  

(17)

which can be fulfilled by the responses of many real problems or dynamical systems, such as the random vibration of cables.

If setting \( A_1 = w_0 \) and the joint PDF of \( w_0 \) and \( \dot{w}_0 \) is needed, the joint PDF of \( A_1, \dot{A}_1, F, \dot{F} \) must be obtained first in the case that \( F(t) \) is governed by Equation (2) in order to keep the diffusion term in the FPK equation as shown in the following solution procedure. In order to obtain the joint PDF of \( A_1, A_1, F, \dot{F} \), separate the state vector \( X \) into two parts as \( X_1 = \{A_1, \dot{A}_1, F, \dot{F}\} = \{X_1, X_2, X_{n_k-1}, X_{n_k}\} \in \mathbb{R}^d \), and \( X_2 \in \mathbb{R}^{n_k-4} \), i.e., \( X = \{X_1, X_2\} \in \mathbb{R}^{n_k} = \mathbb{R}^{n_{x_1}} \times \mathbb{R}^{n_{x_2}} \) with \( n_{x_1} = 4 \) and \( n_{x_2} = n_k - 4 \).

Denote the PDF of \( X_1 \) as \( p_1(x_1) \). In order to obtain \( p_1(x_1) \), integrating both sides of Equation (16) over \( \mathbb{R}^{n_{x_2}} \) gives

\[ \int_{\mathbb{R}^{n_{x_2}}} \frac{\partial}{\partial x_i} [f_i(x)p(x)] dx_2 - \frac{\alpha^2 S}{2} \int_{\mathbb{R}^{n_{x_2}}} \frac{\partial^2 p(x)}{\partial x_i^2} dx_2 = 0 \]  

(18)

Because of the conditions given by Equation (17), we have

\[ \int_{\mathbb{R}^{n_{x_2}}} \frac{\partial}{\partial x_i} [f_i(x)p(x)] dx_2 = 0 \quad x_i \in \mathbb{R}^{n_{x_2}} \]  

(19)
Substituting Equation (24) into Equation (23) and noting Equation (22) gives

\[ \int_{\mathbb{R}^{n_k}} \frac{\partial}{\partial x} \left[ f_i(x) p(x) \right] dx_2 - \frac{\alpha^2 S}{2} \int_{\mathbb{R}^{n_k}} \frac{\partial^2 p(x)}{\partial x^2} dx_2 = 0 \]

\[ x_i \in \mathbb{R}^{n_k} \]  

which can be equivalently written as

\[ \int_{\mathbb{R}^{n_k}} \frac{\partial}{\partial x} \left[ f_i(x) p(x) \right] dx_2 - \frac{\alpha^2 S}{2} \int_{\mathbb{R}^{n_k}} \frac{\partial^2 p(x)}{\partial x^2} dx_2 = 0 \]

\[ x_i \in \mathbb{R}^{n_k} \]  

Because

\[ \int_{\mathbb{R}^{n_k}} p(x) dx_2 = p_1(x) \]  

then Equation (21) can be further written as

\[ \int_{\mathbb{R}^{n_k}} \frac{\partial}{\partial x} \left[ f_i(x) p(x) \right] dx_2 - \frac{\alpha^2 S}{2} \frac{\partial^2 p_1(x)}{\partial x^2} = 0 \]

\[ x_i \in \mathbb{R}^{n_k} \]  

Clustering the terms purely in \( x \) in one part and the other terms in another part, then \( f_i(x) \) is decomposed into two parts as

\[ f_i(x) = f_i^I(x_1) + f_i^{II}(x). \]  

Substituting Equation (24) into Equation (23) and noting Equation (22) gives

\[ \frac{\partial}{\partial x} \left[ f_i^I(x_1) p_1(x_1) + \int_{\mathbb{R}^{n_k}} f_i^{II}(x)p(x)dx_2 \right] - \frac{\alpha^2 S}{2} \frac{\partial^2 p_1(x_1)}{\partial x^2} = 0 \]

\[ x_i \in \mathbb{R}^{n_k} \]  

Denote \( f_i^{II}(x) = \sum_k f_i^{II}(x_1, z_k) \) in which \( z_k \in \mathbb{R}^{n_k} \subset \mathbb{R}^n. n_k \) is the number of the state variables in \( z_k \). Then Equation (25) is written as

\[ \frac{\partial}{\partial x} \left[ f_i^I(x_1) p_1(x_1) + \sum_k \int_{\mathbb{R}^{n_k}} f_i^{II}(x_1, z_k) p_k(x_1, z_k)dx_2 \right] - \frac{\alpha^2 S}{2} \frac{\partial^2 p_1(x_1)}{\partial x^2} = 0 \]

\[ x_i \in \mathbb{R}^{n_k} \]  

in which \( p_k(x_1, z_k) \) denotes the joint PDF of \( \{X_1, Z_k\} \). The summation convention not applies for the indexes \( k \) in Equation (26) and in the following discussions.

From Equation (26), it is seen that the coupling of \( X_1 \) and \( X_2 \) comes from \( f_i^{II}(x_1, z_k) p_k(x_1, z_k) \). Express \( p_k(x_1, z_k) \) as

\[ p_k(x_1, z_k) = p_1(x_1) q_k(z_k; x_1) \]

where \( q_k(z_k; x_1) \) is the conditional PDF of \( Z_k \) for given \( X_1 = x_1 \).

Substituting Equation (27) into Equation (26) gives

\[ \frac{\partial}{\partial x} \left\{ \left[ f_i^I(x_1) + \sum_k \int_{\mathbb{R}^{n_k}} f_i^{II}(x_1, z_k) q_k(z_k; x_1) dx_2 \right] p_1(x_1) \right\} - \frac{\alpha^2 S}{2} \frac{\partial^2 p_1(x_1)}{\partial x^2} = 0 \]

\[ x_i \in \mathbb{R}^{n_k} \]  

Approximately replacing the conditional PDF \( q_k(z_k; x_1) \) by that obtained from EQL, then Equation (28) is written as

\[ \frac{\partial}{\partial x} \left\{ \left[ f_i^I(x_1) + \sum_k \int_{\mathbb{R}^{n_k}} f_i^{II}(x_1, z_k) \tilde{q}_k(z_k; x_1) dx_2 \right] \tilde{p}_1(x_1) \right\} - \frac{\alpha^2 S}{2} \frac{\partial^2 \tilde{p}_1(x_1)}{\partial x^2} = 0 \]

\[ x_i \in \mathbb{R}^{n_k} \]
where \( q_k(z_k; x_1) \) is the conditional PDF of \( Z_k \) obtained from EQL for given \( X_1 = x_1 \) and \( \tilde{p}_1(x_1) \) is the approximate PDF of \( X_1 \). Denote
\[
\tilde{f}_i(x_1) = f^I_i(x_1) + \sum_k \int_{\mathbb{R}^n} f^H_{i,k}(x_1, z_k) \bar{q}_k(z_k; x_1) dz_k
\]
(30)

Then Equation (29) can be finally written as
\[
\frac{\partial}{\partial x_i} \left[ \tilde{f}_i(x_1) \tilde{p}_1(x_1) \right] - \frac{\alpha^2 S}{2} \frac{\partial^2 \tilde{p}_1(x_1)}{\partial x^2_{n_i}} = 0 \quad x_i \in \mathbb{R}^{n_i}
\]
(31)
which is the approximate FPK equation for the joint PDF of the state variables in the subspace \( \mathbb{R}^{n_i} \) or \( \mathbb{R}^4 \). Because the approximate FPK equation (31) is in four dimensions, it is solvable with the EPC method [13]. Hence the whole solution procedure is named SSS-EPC method.

5 Probabilistic solutions of the cable system excited by uniformly distributed filtered Gaussian white noise

Consider the steel cable with Young’s modulus \( E = 2.1 \times 10^{11} N/m^2 \), damping ratio for each mode \( \xi = 0.01 \), length \( l = 120 m \), diameter of the cable cross section being \( 0.1 m \), material density \( \rho = 7,850 kg/m^3 \), sag-to-span ratio \( d/l = 1/500 \), the inclined angle \( \theta = 0^\circ \). The cable is excited by uniformly distributed force with density \( 2 \times 10^4 F(t) \) in which \( F(t) \) is filtered Gaussian white noise governed by Equation (2) with \( \xi_0 = 0.3 \), \( \omega_0 = 7.1 \), \( \alpha = 1 \), and \( S = 0.1 \). The power spectral density of \( F(t) \) is shown in Figure 2. The PDF of the deflection at the center of the cable with \( x_0 = 0.5l \) is analyzed. In this case, the \( X_1 \) is given by \( X_1 = \{w(0.5l,t), \dot{w}(0.5l,t), F(t), \dot{F}(t)\} \) in the SSS dimensionality reduction procedure.

![Figure 2. Power spectral density distribution of the force F(t)](image)

When the number of the shape functions equals one, the formulated equation of motion is a 2DOF system. It is known that the exact PDF of the response of this 2DOF system is not obtainable. The FPK equation governing the stationary PDF is in 4 dimensions and it can be solved directly with EPC method [13]. The PDFs of the deflection at the center of the cable are obtained with the EPC method, MCS, and EQL method, respectively. The approximate PDFs of the deflection obtained with the above three methods are shown and compared in Figure 3(a). The tails of the PDFs obtained with various methods are also compared in Figure 3(b) with the logarithmic value of the PDF of central deflection. In these figures, \( \sigma_w \) denotes the standard deviation of the deflection obtained with EQL at the center of the cable. It is observed in Figures 3(a) and 3(b) that the result obtained with EPC is close to MCS while the result corresponding to EQL deviates a lot from MCS. The sample size in MCS is \( 10^9 \). The polynomial degree used in the EPC solution procedure is four.
When the number of the shape functions increases to 5, the formulated system is of 6 degrees of freedom. The PDFs of the deflection at the center of the cable are obtained with the SSS-EPC method, MCS, and EQL method, respectively. They are shown and compared in Figure 4(a). The tails of the PDFs obtained with various methods are also compared in Figure 4(b) with the logarithmic values of the PDF of central deflection. The simulation about this 6DOF system was conducted on the original 6DOF system rather than on the 2DOF system resulted from the SSS dimensionality reduction procedure. The sample size in MCS is $10^8$. The polynomial degree used in the EPC solution procedure is four when the approximate FPK equation from SSS procedure is solved with EPC method.

Numerical experience showed that further increasing the number of mode functions beyond five can not further change the solution obviously. Hence the solution of this 6DOF system can be considered as converged solution in the sense of Galerkin approximation.

Further increasing the number of samples beyond $10^8$, such as $10^9$, in analyzing the 6DOF system with MCS can make the simulated results smoother and closer to the tails of PDF, but it can also make the computational effort huge due to the huge number of nonlinear terms in the system. There are about 250 nonlinear terms and 50 linear terms in the formulated 6DOF system. Hence only part of the PDF of deflection was obtained with MCS for this nonlinear system due to the limited sample size. It is one of the challenges inherent in MCS. Even so, the computational time needed by SSS-EPC method is about 10 seconds which is mainly spent on the EQL procedure, while the computational time needed by MCS is about 2.5 hours for this 6DOF system in the same computer with Inte(R) CPU B950@2.10GHz, and 3.16GB RAM, and the same running environment. From Figures 3(b) and 4(b) it is observed that there is some difference between the PDF of the deflection obtained by modeling the system as a 2DOF system and the PDF of the deflection obtained by modeling the system as a 6DOF system. The results obtained from 2DOF system is about 0.96 to 0.92 times of the solution of the 6DOF system when the deflection changes from $-3.5\sigma_w$ to $-4\sigma_w$ and about 0.89 to 0.81 times of the solution of the 6DOF system when the deflection changes from $3.5\sigma_w$ to $4\sigma_w$, by comparing the the data for drawing Figures 3(b) and 4(b). For either the 2DOF system or the 6DOF system, the results from EQL are far from being acceptable.

![Figure 3](image-url)  
*(a) PDFs; (b) Logarithm of PDFs*
6 Conclusions

The MDOF-NSD system is formulated for the cable with small sag and excited by uniformly distributed filtered Gaussian white noise governed by a linear second-order stochastic differential equation. The dimension-reduction procedure of state-space-split method is employed to make the high-dimensional FPK equation governing the PDF solution of the cable reduced to the FPK equation in 4-dimensional space. Then the exponential polynomial closure method is adopted to solve the FPK equation in 4-dimensional space. From numerical analysis it is observed that the SSS-EPC method works well for obtaining the PDF of the deflection of the cable in this case. It is observed that the results obtained by modeling the cable as 6DOF system is improved in comparison with those obtained by modeling the cable as 2DOF system. Further increasing the number of degrees of freedom cannot make the solution changed obviously. It means that the solution obtained with this 6DOF system can be considered as converged solution in the sense of approximation with Galerkin method in the given case. It is also found that the computational time needed by MCS for the 6DOF system is about 900 times of that needed by the SSS-EPC method even if the sample size is limited to $10^8$. The solution from EQL is not acceptable for this nonlinear random vibration problem of cable.

Acknowledgement

The results presented in this paper were obtained under the supports of the Research Committee of University of Macau (Grant No. MYRG138-FST11-EGK, MYRG2014-0084-FST) and the Science and Technology Development Fund of Macau (Grant No. 043/2013/A).

REFERENCES


Numerical integration of underactuated mechanical systems subjected to mixed holonomic and servo constraints

Peter Betsch*, Robert Altmann#, Yinping Yang*

* Institut für Mechanik
Karlsruher Institut für Technologie
Kaiserstraße 12, 76131 Karlsruhe, Germany
peter.betsch@kit.edu, yinping.yang@kit.edu

# Institut für Mathematik
Technische Universität Berlin
Straße des 17. Juni 136, 10623 Berlin, Germany
raltmann@math.tu-berlin.de

ABSTRACT
A new index reduction approach is developed for the inverse dynamics simulation of underactuated mechanical systems. The underlying equations of motion contain both holonomic and servo constraints. The proposed method is applied to a very general and versatile formulation of cranes. The numerical results demonstrate the functional efficiency of the method.

Keywords: Underactuated mechanical systems, feedforward control, inverse dynamics, differentially flat systems.

1 INTRODUCTION
We present a new approach to the inverse dynamics simulation of discrete mechanical systems. The proposed method is relying on the use of servo constraints for the partial specification of the motion of mechanical systems (see, for example, [1, 2, 3]). In particular, we focus on the specification of trajectories of specific points of a multibody system such as the end effector of a robot.

The partial specification of the motion of a multibody system by means of servo constraints typically leads to a problem formulation in terms of differential-algebraic equations (DAEs). If minimal coordinates are used, the differential part of the DAEs corresponds to the equations of motion while the algebraic part is related to the servo constraints. The servo constraints enforce the desired motion along prescribed trajectories and thus specify the control outputs of the system. To determine the associated control inputs required to steer the system such that the prescribed trajectories are tracked, the DAEs need to be solved. In this way, a simulation approach to the feedforward control of multibody systems can be realized.

In the special case of fully actuated multibody systems the simulation approach to the inverse dynamics problem yields index-3 DAEs that can be integrated in analogy to the DAEs corresponding to constrained mechanical systems (see, for example, [4]). However, the situation changes considerably if underactuated mechanical systems are dealt with. In this type of systems the number of degrees of freedom exceeds the number of controls. Examples of underactuated systems are cranes and flexible multibody systems. The use of servo constraints in the context of underactuated multibody systems leads to a broad diversity of servo constraint problems (see, in particular, the recent papers [5, 6, 7]). One indicator of problem diversity is the (differentiation) index of the underlying DAEs that typically ranges from three to five and even higher. Consequently, to facilitate a stable numerical integration some kind of index reduction approach needs to be applied.

In the present work we apply a specific index reduction technique called minimal extension (see [8]). Index reduction by minimal extension is based on the introduction of new algebraic variables...
along with the enlargement of the DAEs by appending time derivatives of the constraints. In our previous work [9] we have shown that index reduction by minimal extension can be applied very efficiently by exploiting the specific structure provided by underactuated mechanical systems. In this connection either minimal coordinates or redundant coordinates can be used. We verified that our approach is a viable alternative to the projection method developed in [10].

We focus on the inverse dynamics of a family of crane models that are known to belong to the class of differentially flat systems. We have shown in our previous work [9] that in a first step the minimal extension approach can be used to lower the index of the DAEs from five to three. In a second step the index can even be reduced to one.

The goal of the present work is to extend the applicability of our index reduction approach to mechanical models of underactuated systems that rely on arbitrarily selected redundant coordinates. Specifically, in contrast to our previous work [9], the number of holonomic constraints is not limited. Consequently, general crane formulations such those developed in [11] can now be included into the present index reduction approach. Similarly, other rotationless formulations of multibody dynamics such as natural coordinates or Cosserat-type descriptions in terms of directors (including rigid bodies and nonlinear beams and shells) typically yield a large number of holonomic constraints. These formulations are now embraced as well by our newly developed index reduction method.

An outline of the rest of our paper is as follows. In Section 2 we introduce the general description of mechanical systems subjected to both holonomic and servoconstraints. In Section 3 we present index reduction by minimal extension and link the present formulation to our previous work [9]. After the discretization in time of the present DAEs in Section 4, a sample application is dealt with in Section 5. Eventually, conclusions are drawn in Section 6.

2 INVERSE DYNAMICS OF UNDERACTUATED MECHANICAL SYSTEMS

We start with a general formulation of mechanical systems subjected to both holonomic and servo constraints. In particular, we consider equations of motion of the form

\[
\begin{bmatrix}
M_1(\bar{p}) & 0 \\
0 & M_2
\end{bmatrix}
\begin{bmatrix}
\ddot{\bar{p}} \\
\ddot{x}
\end{bmatrix}
= \begin{bmatrix}
\bar{f}_1(\bar{p}, \bar{p}) \\
\bar{f}_2(x, \dot{x})
\end{bmatrix} + \begin{bmatrix}
\bar{B}_1(\bar{p}) \\
0
\end{bmatrix} \bar{u} - \begin{bmatrix}
\bar{G}_1(\bar{p}, x) \\
\bar{G}_2(\bar{p}, x)
\end{bmatrix} \bar{\lambda},
\]

\tag{1a}
\]

\begin{align}
\bar{h}(\bar{p}) &= 0, \\
\bar{g}(\bar{p}, x) &= 0, \\
x &= \gamma.
\end{align}

\tag{1b, 1c, 1d}

The first row block in (1a) corresponds to the robot (or input) subsystem with coordinates \( \bar{p} \in \mathbb{R}^{\bar{n}-\bar{m}} \), whereas the second row block in (1a) corresponds to the output subsystem with coordinates \( x \in \mathbb{R}^n \). The \( \bar{n} \) redundant coordinates

\[
\bar{q} = \begin{bmatrix}
\bar{p} \\
x
\end{bmatrix}
\]

\tag{2}

are subject to the holonomic constraints (1b) and (1c), with associated constraint functions \( \bar{h} \in \mathbb{R}^{\bar{n}-\bar{m}} \) and \( \bar{g} \in \mathbb{R}^m \). The corresponding constraint Jacobian is given by

\[
\bar{G} = [\bar{G}_1, \bar{G}_2] \in \mathbb{R}^{m,\bar{n}}
\]

\tag{3}

where

\[
\bar{G}_1 = \begin{bmatrix}
D_1\bar{h}(\bar{p}) \\
D_1\bar{g}(\bar{p}, x)
\end{bmatrix} \quad \text{and} \quad \bar{G}_2 = \begin{bmatrix}
0 \\
D_2\bar{g}(\bar{p}, x)
\end{bmatrix}
\]

\tag{4}

Here, \( D_1\bar{g}(\bar{p}, x) \in \mathbb{R}^{m,\bar{n}-\bar{m}} \) denotes the partial derivative with respect to \( \bar{p} \). Similarly, \( D_2\bar{g}(\bar{p}, x) \in \mathbb{R}^{m,\bar{m}} \) denotes the partial derivative with respect to \( x \). Associated to the \( \bar{m} \) holonomic constraints there are Lagrange multipliers \( \bar{\lambda} \in \mathbb{R}^{\bar{m}} \).
Due to the presence of holonomic constraints the equations of motion assume the form of differential-algebraic equations (DAEs). The configuration space of the constrained mechanical system under consideration is defined by

\[ Q = \{ \mathbf{q} \in \mathbb{R}^n | \mathbf{h}(\mathbf{p}) = 0, \dot{\mathbf{g}}(\mathbf{p}, \mathbf{x}) = 0 \} \]  

(5)

Throughout this work we assume that the constraints are independent. Consequently, the constraint Jacobian \( \mathbf{G} \) has full row rank and the discrete mechanical system under consideration has \( n - m \) degrees of freedom.

The servo constraints (1d) specify the desired trajectory of the load via the prescribed function \( \mathbf{y} : [0, t_f] \rightarrow \mathbb{R}^a \), where \( [0, t_f] \) is the time interval of interest. In the present work we focus on underactuated mechanical systems in which the number of controls is smaller than the number of degrees of freedom, that is, \( a < n - m \).

The control inputs \( \mathbf{u} \in \mathbb{R}^a \) regulate the control forces acting on the first subsystem. In this connection \( \mathbf{B}_1 \in \mathbb{R}^{n-a} \) denotes the input transformation matrix. Besides the constraint and control forces, additional forces acting on the system are contained in the conjugate force vectors \( \mathbf{f}_1 \in \mathbb{R}^{n-a} \) and \( \mathbf{f}_2 \in \mathbb{R}^a \). Similarly, the mass matrix is split into the submatrices \( \mathbf{M}_1 \in \mathbb{R}^{n-a,n-a} \) and \( \mathbf{M}_2 \in \mathbb{R}^{a,a} \).

Due to the presence of servo constraints the (differentiation) index of the DAEs (1) often exceeds 3. For example, the application to (differentially flat) crane systems the index is typically reduced from 5 to 3. A rigorous indication to the application of a numerical integrator the index of the DAEs should be lowered. For that purpose, following our previous work [9], we apply index reduction by minimal extension to the DAEs (1).

We emphasize at this point that in the above formulation the number of holonomic constraints, \( m \), is just restricted by \( m < \pi \). This facilitates the arbitrary selection of redundant coordinates best suited for the description and numerical simulation of the specific inverse dynamics problem at hand. This is in contrast to our previous work [9], where we assumed \( m \leq a \).

### 3 INDEX REDUCTION BY MINIMAL EXTENSION

Guided by our previous work [9] we enlarge the system of DAEs (1) by appending the first and second time derivative of the servo constraints. To maintain a square system we introduce additional dummy variables \( \mathbf{\hat{x}} := \dot{\mathbf{x}} \) and \( \mathbf{\hat{\hat{x}}} := \ddot{\mathbf{x}} \). Accordingly, we arrive at

\[
\begin{bmatrix}
\mathbf{M}_1(\mathbf{p}) & 0 \\
0 & \mathbf{M}_2
\end{bmatrix}
\begin{bmatrix}
\ddot{\mathbf{p}} \\
\mathbf{\hat{x}}
\end{bmatrix}
= \begin{bmatrix}
\mathbf{f}_1(\mathbf{p}, \mathbf{p}) \\
\mathbf{f}_2(\mathbf{x}, \mathbf{\hat{x}})
\end{bmatrix}
+ \begin{bmatrix}
\mathbf{B}_1^T(\mathbf{p}) \\
0
\end{bmatrix}
\mathbf{u}
- \begin{bmatrix}
\mathbf{G}_1^T(\mathbf{p}, \mathbf{x}) \\
\mathbf{G}_2^T(\mathbf{p}, \mathbf{x})
\end{bmatrix}
\mathbf{\hat{\hat{x}}},
\]

(6a)

\[
0 = \mathbf{h}(\mathbf{p}),
\]

(6b)

\[
0 = \mathbf{g}(\mathbf{p}, \mathbf{x}),
\]

(6c)

\[
\mathbf{x} = \mathbf{y},
\]

(6d)

\[
\dot{\mathbf{x}} = \mathbf{\gamma},
\]

(6e)

\[
\mathbf{\hat{x}} = \mathbf{\gamma}.
\]

(6f)

Provided that certain assumptions apply the minimally extended DAEs (6) have index 3. In particular, for differentially flat crane systems the index is typically reduced from 5 to 3. A rigorous proof of that fact can be found in our previous work [9].

#### 3.1 Reduction of the number of redundant coordinates

To link the DAEs (6) to the formulation in [9] we reduce the number of redundant coordinates to \( n < \pi \) by introducing the mapping \( \mathbf{\varphi} : \mathbb{R}^{n-a} \rightarrow \mathbb{R}^{\pi-a} \) given by

\[
\mathbf{\varphi} = \mathbf{\varphi}(\mathbf{p})
\]

(7)
such that the constraints (6b) are identically satisfied for arbitrary $p \in \mathbb{R}^{n-a}$. Consequently,

$$\overline{h}(\varphi(p)) = 0 \quad \text{and} \quad D\overline{h}(\overline{p})D\varphi(p) = 0$$ (8)

for $\overline{p} = \varphi(p)$. Premultiplying the first row block in (6a) by $D\varphi^T(p)$ and taking into account (7) and (8) yields the size-reduced DAEs

$$\begin{bmatrix} M_1(p) & 0 \\ 0 & M_2 \end{bmatrix} \begin{bmatrix} \dot{p} \\ \dot{x} \end{bmatrix} = \begin{bmatrix} f_1(p, \dot{p}) \\ f_2(x, \dot{x}) \end{bmatrix} + \begin{bmatrix} B_1^T(p) \\ B_2^T(p) \end{bmatrix} u - \begin{bmatrix} D_1 g_1^T(p, x) \\ D_2 g_2^T(p, x) \end{bmatrix} \lambda,$$ (9a)

$$0 = g(p, x),$$ (9b)

$$x = \gamma,$$ (9c)

$$\dot{x} = \gamma,$$ (9d)

$$\dot{\bar{x}} = \gamma,$$ (9e)

where

$$M_1(p) = D\varphi^T(p)\overline{M}_1(\overline{p})D\varphi(p)$$ (10a)

$$f_1(p, \dot{p}) = D\varphi^T(p)\left(\overline{f}_1(\overline{p}, \overline{\varphi}) - \overline{M}_1(\overline{p})\frac{d}{dt}(D\varphi(p))\dot{\varphi}\right)$$ (10b)

$$B_1^T(p) = D\varphi^T(p)\overline{B}_1^T(\overline{p})$$ (10c)

$$g(p, x) = \overline{\gamma}(\overline{p}, x)$$ (10d)

for $\overline{p} = \varphi(p)$. In this way the number of redundant coordinates is reduced by $\overline{m} - m$ such that the remaining coordinates are given by the $n$-dimensional configuration vector

$$q = \begin{bmatrix} p \\ x \end{bmatrix}$$ (11)

Note that (9b) contain the $m$ remaining holonomic constraints with associated Lagrange multipliers $\lambda \in \mathbb{R}^m$ in (9a). The configuration space of the constrained mechanical system under consideration can now be expressed in the form

$$Q = \{q \in \mathbb{R}^n | g(p, x) = 0\}$$ (12)

The DAEs (9) lie at the heart of our previous work [9]. There it is shown that under certain conditions (e.g. $m \leq a$ and $M_1(p)$ non-singular) the minimally extended DAEs (9) attain an index reduction by two. In the case of differentially flat crane models the original DAEs have index 5, whereas the index-reduced DAEs (9) (and, correspondingly, the DAEs (6) as well) have index 3. It is further shown in [9] that a second application of index reduction by minimal extension can achieve a reduction to index-1 DAEs.

### 4 DISCRETIZATION

For the specific inverse dynamics problem dealt with in the next section the proposed index reduction approach yields DAEs (6) with index 3. Due to the semi-explicit form of the DAEs (6) we can expect the simple Euler-backward discretization to work well (see Ascher & Petzold [12, Sec. 10.1.1]). The DAEs (6) can be recast in the form

$$\overline{M}_1(\overline{p})\dot{\overline{p}} - \overline{f}_1(\overline{p}, \overline{\varphi}) + \overline{B}_1^T(\overline{p})u - \overline{G}_1^T(\overline{p}, \gamma)\overline{\lambda}$$ (13a)

$$0 = M_2\dot{\gamma} - f_2(\gamma, \dot{\gamma}) + G_2^T(\overline{p}, \gamma)\overline{\lambda}$$ (13b)

$$0 = \overline{\gamma}(\overline{p}, \gamma)$$ (13c)

$$0 = \overline{h}(\overline{p})$$ (13d)
The DAEs (13) provide \( \pi - a \) differential equations (13a) along with \( a + m \) algebraic equations (13b) through (13d) for the determination of \( \mathbf{p} \in \mathbb{R}^{\pi - a}, \mathbf{u} \in \mathbb{R}^a, \) and \( \mathbf{x} \in \mathbb{R}^m. \) Application of the Euler-backward method yields

\[
\begin{align*}
\overline{\mathbf{p}}_{n+1} - \overline{\mathbf{p}}_n &= \Delta t \overline{\mathbf{v}}_{n+1} \\
\mathbf{M}_1(\overline{\mathbf{p}}_{n+1}) \left( \frac{(\overline{\mathbf{v}}_{n+1} - \overline{\mathbf{v}}_n)}{\Delta t} \right) &= \mathbf{f}_1(\overline{\mathbf{p}}_{n+1},\overline{\mathbf{v}}_{n+1}) + \mathbf{B}_1^T(\overline{\mathbf{p}}_{n+1})\overline{\mathbf{v}}_{n+1} - \mathbf{G}_1^T(\overline{\mathbf{p}}_{n+1},\mathbf{y}(t_{n+1}))\overline{\mathbf{x}}_{n+1} \\
0 &= \mathbf{M}_2(\mathbf{y}(t_{n+1}) - \mathbf{f}_2(\mathbf{y}(t_{n+1}),\mathbf{y}(t_{n+1}))) + \mathbf{G}_2^T(\mathbf{p}_{n+1},\mathbf{y}(t_{n+1}))\overline{\mathbf{x}}_{n+1} \\
0 &= \mathbf{h}(\overline{\mathbf{p}}_{n+1}) \\
0 &= \mathbf{h}(\overline{\mathbf{p}}_{n+1})
\end{align*}
\tag{14a}
\tag{14b}
\tag{14c}
\tag{14d}
\tag{14e}
\]

In a typical time step of size \( \Delta t = t_{n+1} - t_n \) we seek approximations \((\bullet)_{n+1}\) to \((\bullet)(t_{n+1})\) given the corresponding quantities \((\bullet)_{n}\) as result of the previous step. For the initial time step we require consistent initial values \( \overline{\mathbf{p}}_0 \) and \( \overline{\mathbf{v}}_0 \) that have to satisfy \( \mathbf{f}(\mathbf{p}_0,\mathbf{y}(t_0)) = 0 \) and \( \mathbf{h}(\mathbf{p}_0) = 0 \) along with

\[
\begin{align*}
\mathbf{D}_1 \mathbf{f}(\mathbf{p}_0,\mathbf{y}(t_0))\overline{\mathbf{v}}_0 + \mathbf{D}_2 \mathbf{f}(\mathbf{p}_0,\mathbf{y}(t_0))\mathbf{y}(t_0) &= \mathbf{0} \\
\mathbf{D}_1 \mathbf{h}(\mathbf{p}_0)\overline{\mathbf{v}}_0 &= \mathbf{0}
\end{align*}
\tag{15a}
\tag{15b}
\]

The scheme (14) provides \( 2\pi + m - a \) algebraic equations for the determination of \( \overline{\mathbf{p}}_{n+1}, \overline{\mathbf{v}}_{n+1} \in \mathbb{R}^{\pi - a}, \mathbf{u}_{n+1} \in \mathbb{R}^a, \) and \( \overline{\mathbf{x}}_{n+1} \in \mathbb{R}^m. \)

5 SAMPLE APPLICATION

![Figure 1. The three-dimensional rotary crane in terms of \( \pi = 10 \) redundant coordinates.](image)

We demonstrate the present approach with the inverse dynamics simulation of a three-dimensional rotary crane. Previously, the crane under consideration has been formulated in terms of \( n = 6 \)}
redundant coordinates (see [13, 9]) or 5 minimal coordinates (see [14, 9]). Alternatively, we now make use of the much more general framework for the modeling of cranes in [11]. In the case of the rotary crane this implies using \( \bar{n} = 10 \) redundant coordinates subjected to \( \bar{m} = 5 \) holonomic constraints. The enlarged set of redundant crane coordinates is given by (Fig. 1)

\[
\bar{p} = \begin{bmatrix} x_2 & y_2 & x_0 & y_0 & L_1 & L_2 & L_0 \end{bmatrix}^T
\]

and

\[
x = \begin{bmatrix} x & y & z \end{bmatrix}^T
\]

The last equation specifies the load (mass \( m \)) coordinates relative to a Cartesian inertial frame. The load is connected to the hoisting winch 2 (Cartesian coordinates \( x_2, y_2, z_2 = 0 \)), actuating torque \( M_2 \), radius \( r_2 \), moment of inertia \( J_2 \) via a rope of length \( L_2 \). The position of the trolley (Cartesian coordinates \( x_0, y_0, z_0 = 0 \), mass \( m_0 \)) on the girder relative to the hoisting winch is given by \( L_0 \). The trolley contains a pulley (radius \( r_w \), moment of inertia \( J_w \)) and is moved along the girder under the action of a second winch 1 (Cartesian coordinates \( x_1, y_1, z_1 = 0 \), actuating torque \( M_1 \), radius \( r_1 \), moment of inertia \( J_1 \)) whose position on the girder relative to the hoisting winch 2 is fixed by the parameter \( \alpha = \frac{1}{2} \). The distance between winch 1 and the trolley is given by \( L_1 \). The holonomic constraints \( \bar{h}(\bar{p}) = 0 \) are given by

\[
\bar{h}(\bar{p}) = \begin{bmatrix}
\frac{1}{2}((x_0 - \alpha x_2)^2 + (y_0 - \alpha y_2)^2 - L_2^2)
\frac{1}{2}((x_0 - x_2)^2 + (y_0 - y_2)^2 - L_0^2)
\frac{1}{2}(x_2^2 + y_2^2 - r_2^2)
\end{bmatrix}
\]

Note that the first two constraints link the coordinates \( L_1 \) and \( L_0 \) to the position of the trolley and, respectively, winch 1 and winch 2. Moreover the third constraint links the parameter \( r \) to the position of winch 2, and the fourth constraint confines the relative motion of the trolley to the longitudinal direction along the girder. The last holonomic constraint \( \bar{g}(\bar{p}, x) = 0 \) is specified by

\[
\bar{g}(\bar{p}, x) = \frac{1}{2}((x - x_0)^2 + (y - y_0)^2 + z^2 - (L_2 - L_0)^2)
\]

and connects the load coordinates with the robot (or crane) coordinates. The total kinetic energy of the mechanical system under consideration assumes the form

\[
T = \frac{1}{2} \bar{p} \cdot \bar{M}_1 \ddot{\bar{p}} + \frac{1}{2} \bar{x} \cdot \bar{M}_2 \ddot{\bar{x}}
\]

in which the mass matrices corresponding to the robot coordinates and the load coordinates are given by

\[
\bar{M}_1 = \begin{bmatrix}
M & M & m_0 \\
M & m_0 & -\frac{J_1}{r_1^2} + \frac{J_2}{r_2^2} - \frac{J_2}{r_2^2} + \frac{J_1}{r_1^2} & -\frac{J_1}{r_1^2}
\end{bmatrix}, \quad \bar{M}_2 = \begin{bmatrix}
m & m \\
m & m
\end{bmatrix}
\]

Here, the mass \( M \) is connected to the moment of inertia of the bridge relative to the z-axis, \( J_b \), via
Further quantities needed in (1) are given by
\[
\mathcal{B}_1^T = \begin{bmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
\frac{1}{\tau_1} & 0 & 0 & 0 \\
0 & \frac{1}{\tau_1} & 0 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix}, \quad \mathcal{J}_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad f_2 = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix}, \quad \mathcal{u} = \begin{bmatrix} M_1 \\ M_2 \\ M_b \end{bmatrix}
\]
and
\[
\mathcal{G}_1^T = \begin{bmatrix}
x_0 - x_2 & \alpha (x_0 - x_2) & 0 & -x_2 & -y_0 \\
y_0 - y_2 & \alpha (y_0 - y_1 y_2) & 0 & -y_2 & x_0 \\
x_2 - x_0 & \alpha x_2 - x_0 & x - x_0 & 0 & y_2 \\
y_2 - y_0 & \alpha y_2 - y_0 & y - y_0 & 0 & -x_2 \\
0 & L_1 & 0 & 0 & 0 \\
0 & 0 & L_2 - L_0 & 0 & 0 \\
0 & 0 & L_0 - L_2 & 0 & 0
\end{bmatrix}, \quad \mathcal{G}_2^T = \begin{bmatrix} 0 & 0 & x_0 - x & 0 & 0 \\
0 & 0 & y_0 - y & 0 & 0 \\
0 & 0 & -z & 0 & 0
\end{bmatrix}
\]

Note that we have \(a = 3\) control inputs given by the two winch torques \(M_1, M_2\), along with the torque \(M_b\) acting about the \(z\)-axis of the rotary crane.

### 5.1 Reduction of the number of redundant coordinates

![Figure 2](image-url)

Figure 2. The three-dimensional rotary crane in terms of a reduced set of \(n = 6\) redundant coordinates.

We next link the above formulation of the rotary crane to the original one (see [13, 9]) relying on the reduced set of crane coordinates (Fig. 2)

\[
p = [\varphi \quad s \quad l]^T
\]

Here, the angle \(\varphi\) measures the rotation of the girder about the \(z\)-axis relative to the \(x\)-axis, \(s\) specifies the position of the trolley on the girder, and \(l\) denotes the length of the hoisting rope.
connecting the load with the winch contained in the trolley. In contrast to the previous crane model the winch contained in the trolley is now assumed to be actuated (torque $M_w$). The previous crane coordinates $\mathbf{p}$ in (16) can now be expressed in terms of the reduced set of crane coordinates (18) giving rise to the mapping $\mathbf{p} = \varphi(\mathbf{p})$ in (7). Accordingly,

\[
\begin{bmatrix}
  x_2 \\
  y_2 \\
  x_0 \\
  y_0 \\
  L_1 \\
  L_2 \\
  L_0
\end{bmatrix} =
\begin{bmatrix}
  -r \cos \phi \\
  -r \sin \phi \\
  s \cos \phi \\
  s \sin \phi \\
  s + \alpha r \\
  s + r + l \\
  s + r
\end{bmatrix}
\]

and $D\varphi(\mathbf{p}) =$

\[
\begin{bmatrix}
  r \sin \phi & 0 & 0 \\
  -r \cos \phi & 0 & 0 \\
  -s \sin \phi & \cos \phi & 0 \\
  s \cos \phi & \sin \phi & 0 \\
  0 & 1 & 0 \\
  0 & 1 & 0 \\
  0 & 1 & 0
\end{bmatrix}
\]

Furthermore, the quantities in (10) can now be calculated in a straightforward way leading to

\[
M_1(\mathbf{p}) =
\begin{bmatrix}
  Mr^2 + m_0 s^2 & 0 & 0 \\
  0 & m_0 + \frac{L_2}{r_1^2} + \frac{L_2}{r_2^2} & \frac{L_2}{r_1^2} \\
  0 & \frac{L_2}{r_2^2} & \frac{L_2}{r_1^2} + \frac{L_2}{r_2^2}
\end{bmatrix}
\]

and

\[
f_1(\mathbf{p}, \dot{\mathbf{p}}) =
\begin{bmatrix}
  -2m_0 s \dot{s} \phi \\
  m_0 s \dot{\phi}^2 \\
  0
\end{bmatrix},

B_1^T(\mathbf{p}) =
\begin{bmatrix}
  0 & 0 & 1 \\
  1 & \frac{1}{r_1} & \frac{1}{r_2} \\
  0 & \frac{1}{r_1} & 0
\end{bmatrix}
\]

There remains one holonomic constraint (9b) which is given by

\[
g(\mathbf{p}, \mathbf{x}) = \frac{1}{2}((x - s \cos \phi)^2 + (y - s \sin \phi)^2 + z^2 - l^2) = 0
\]

We finally remark that the control inputs $\mathbf{u} := [M_b, F_t, M_w/r_w]^T$ conjugated to the reduced crane coordinates (18) can be obtained from $\mathbf{u} = B_1^T(\mathbf{p})\mathbf{\dot{u}}$. In particular,

\[
F_t = \frac{M_1}{r_1} + \frac{M_2}{r_2}
\]

\[
M_w = \frac{r_w}{r_2} M_2
\]

That is, the two winch torques $M_1$ and $M_2$ of the original model are linked to the force $F_t$ acting on the trolley and the winch torque $M_w$ (cf. Fig. 2).

### 5.2 Numerical results

In the numerical simulation we make use of the same data as in [14, 9]. In particular, the inertia parameters are given by $m = 100$, $m_0 = 10$, $J_b = 480$, and $M = \frac{1}{10}$ with $r = 4$. Concerning the moment of inertia corresponding to the winches, we choose $J_w = 0.1$, and $J_1 = J_2 = 0$. Moreover, $r_w = 0.1$. The servo constraints are used to prescribe a rest-to-rest maneuver of the load specified by $\mathbf{y}(t) = \mathbf{y}_0 + (\mathbf{y}_f - \mathbf{y}_0)c(t)$ with $\mathbf{y}_0 = [5 \ 0 \ -5]$ at $t_0 = 0$ and $\mathbf{y}_f = [-2 \ 2 \ -2]$ at $t_f = 20$. The function $c(t)$ is composed of three phases,

\[
c(t) =
\begin{cases}
  c_I(t) & \text{for } 0 \leq t < 5, \\
  c_{II}(t) & \text{for } 5 \leq t < 15, \\
  c_{III}(t) & \text{for } 15 \leq t \leq 20
\end{cases}
\]

116
with

\[ c_1(t) = \frac{1}{\tau - \tau_0} \left( -\frac{5t^8}{2\tau_0^6} + \frac{10t^7}{\tau_0^5} - \frac{14t^6}{2\tau_0^4} + \frac{7t^5}{2\tau_0^3} \right), \]

\[ c_2(t) = \frac{1}{\tau - \tau_0} \left( t - \tau_0 \right), \]

\[ c_3(t) = 1 + \frac{1}{\tau - \tau_0} \left( -\frac{5(t - t)^8}{2\tau_0^6} + \frac{10(t - t)^7}{\tau_0^5} - \frac{14(t - t)^6}{2\tau_0^4} + \frac{7(t - t)^5}{2\tau_0^3} \right). \]

Using the reduced crane coordinates, the initial configuration of the rotary crane at \( t_0 = 0 \) is defined by \( p_0 = \begin{bmatrix} 0 & 5 & 5 \end{bmatrix}^T \), while the initial load coordinates are given by \( x_0 = \begin{bmatrix} 5 & 0 & -5 \end{bmatrix}^T \).

In Figs. 3 and 4, the numerical solution (NUM) is compared to the analytical reference solution (REF) obtained in [9]. It can be observed that the numerical solution converges to the reference solution when the time step size is reduced. In addition to that Fig. 5 displays the numerical solution for the extended crane coordinates \( \mathbf{p} \). The two alternative formulations in terms of redundant coordinates (\( \mathbf{p} \) and \( \mathbf{p} \), respectively) yield practically indistinguishable results. The simulated motion of the rotary crane in terms of extended crane coordinates \( \mathbf{p} \) is illustrated in Fig. 6 with some snapshots at consecutive points in time. Similarly, snapshots obtained with the formulation in terms of the reduced crane coordinates \( \mathbf{p} \) are shown in Fig. 7.

**Figure 3.** Rotary crane: Comparison between the numerical results (NUM) obtained with \( \Delta t = 1 \) and the reference solution (REF) for the reduced crane coordinates \( p(t) \).
Figure 4. Rotary crane: Comparison between the numerical results (NUM) obtained with $\Delta t = 0.1$ and the reference solution (REF) for the reduced crane coordinates $\mathbf{p}(t)$.

Figure 5. Rotary crane: Numerical results (NUM) for the extended crane coordinates $\mathbf{p}(t)$ obtained with $\Delta t = 0.1$. 
**Figure 6.** Rotary crane (formulation in terms of the extended crane coordinates $\mathbf{\bar{p}}$): Snapshots at specific points in time.

**Figure 7.** Rotary crane (formulation in terms of the reduced crane coordinates $\mathbf{p}$): Snapshots at specific points in time.
6 CONCLUSIONS

We have dealt with a general formulation of underactuated mechanical systems subjected to both holonomic and servo constraints. It has been shown that index reduction by minimal extension makes possible the numerically stable inverse dynamics simulation of such systems. In contrast to our previous work [9] the present formulation does not restrict the level of redundancy of the coordinates used to model a specific mechanical system. Accordingly, truly rotationless formulations of multibody systems such as natural coordinates are now encompassed by our index reduction approach. The functional efficiency of the newly proposed method has been demonstrated in the framework of a general formulation of cranes. The corresponding rotationless formulation is characterized by constant inertia parameters and holonomic constraints that are at most quadratic in the coordinates. In forward dynamics, this type of formulation makes possible the design of structure-preserving integrators. Structure-preserving integrators for the inverse dynamics simulation of underactuated mechanical systems are not available yet. The present formulation is deemed to provide a good foundation for the development of such integrators which will be the goal of future work.

REFERENCES


Relationship Between the Natural and Relative Coordinates Through the DeNOC Matrices

Anil K. Sharma, Subir K. Saha
Department of Mechanical Engineering
Indian Institute of Technology Delhi,
New Delhi- 110016, India
agroupofindia@gmail.com, saha@mech.iitd.ac.in

ABSTRACT
In dynamics of the multibody systems, the selection of coordinates is important. Different types of coordinate systems are available for dynamic formulation and have their own advantages and disadvantages. Here, a relationship between natural and relative coordinates is developed through the DeNOC matrices. The DeNOC matrices are used to develop a transformation matrix. This transformation matrix transforms the dynamic formulation from natural to relative coordinates. First, the dynamic formulation is done using the natural coordinates. Then a transformation matrix is developed to transform the dynamic formulation into relative coordinates. The transformation matrices are developed for the four-bar mechanism and 3-RRR parallel manipulator. The results for both the mechanism are compared in terms of CPU time, Constraints violation and Energy drift.

Keywords: DeNOC matrices, Dynamic analyses, Natural and relative coordinates, Velocity transformation.

1 INTRODUCTION
At present, the number of methods are available for the derivation of equations of motion for the multibody systems. These methods can be divided into two main approaches. In the first approach, the equations of motion are derived using the relative joint coordinates [1], [2], [5] and [7]. Open-loop multibody systems can be specified using the independent generalized coordinates, while for the closed-loop systems a minimum number of dependent coordinates should be defined. Dynamic formulation using the relative joint coordinates leads to an efficient solution and integration of the equations of motion. However, in many applications, this approach leads to a relatively complex recursive formulation based on the loop closure equations.

In the second approach, the equations of motion are derived using the natural coordinates [3], [8] and [18]. These coordinates describe the absolute location and orientation of the bodies with respect to the inertial frame. The dynamic formulation is straightforward and the kinematic constraints equations are easily formulated. The dynamic formulation of multibody systems based on natural and joint coordinates was done using velocity transformation in [10]. The equations of motion can be formulated easily in large number of natural coordinates (absolute accelerations), however the drawback is the large number of mixed differential-algebraic equations. The numerical solution of these equations is computationally inefficient. On the other hand, the dynamic formulation using the relative coordinates is cumbersome, while they are computationally efficient. Further, the extra effort is required for the computation of the absolute positions, velocities, and accelerations of the multibody systems. A systematic method to derive the minimal set of equations of motion was presented in [14]. The equations of motion were written first in dependent coordinates, then velocity transformation matrix was used to derive the minimal set of equations of motion.

Note that an efficient recursive dynamic formulation, using the DeNOC matrices was developed in [5]. This tempted us to develop a similar set of the DeNOC matrices relating the time-derivatives of the natural coordinates and those of the relative coordinates to obtain a set of
required independent equations of motion. A relationship between the natural and relative coordinates was developed at velocity level. We believe that this yields in minimal set of differential equations of motion and provide an efficient numerical solution. The dynamic formulation is demonstrated using two mechanism, namely four-bar mechanism and 3-RRR parallel manipulator. The velocity transformation matrix is developed for the two link system and triangular platform, as per the requirement of the sub-systems of the above mechanisms.

2 VELOCITY TRANSFORMATION MATRIX

The velocity transformation matrix is developed using the DeNOC matrices. The DeNOC matrices are taken from [5]. First, the velocity transformation matrix is developed for the two link systems and then for the triangular link.

2.1 Two Link System

The two link serial manipulator is shown in Figure 1. The \( \mathbf{a}_i \) is a vector between two adjacent joints (link length vector \( \mathbf{a}_i = \mathbf{d}_i + \mathbf{r}_i \)), \( \mathbf{c}_i \) is the vector from the origin of the inertial frame to the center-of-mass of the \( i \)th link, \( \mathbf{d}_i \) is a vector from the origin of the link to the center-of-mass of the link, and \( \mathbf{r}_i \) is the vector form the center-of-mass of the link to the origin of the adjacent link as shown in Figure 1. where \( i = 1, 2 \). The generalized equation of motion using the Lagrange formulation can be written in terms of natural coordinates as

\[
\mathbf{I}_n \ddot{\mathbf{v}} + \mathbf{J}^T \lambda = \mathbf{\emptyset} (1)
\]

Where \( (\mathbf{I}_n) \) is the Generalized Inertia Matrix (GIM) in natural coordinates, \( \mathbf{J} \) is the constraint Jacobian matrix, \( \lambda \) is the Lagrange multiplier, \( \mathbf{\emptyset} \) is the Generalized Force Vector (GFV), and \( \mathbf{v} \) is the vector of linear velocities of the joints. The \( 4 \times 4 \) GIM, Jacobian matrix and the GFV is given below:

\[
\mathbf{I}_n = \begin{bmatrix}
\frac{(m_1 + m_2)}{3} & 0 & \frac{m_2}{6} & 0 \\
0 & \frac{(m_1 + m_2)}{3} & 0 & \frac{m_2}{6} \\
\frac{m_2}{6} & 0 & \frac{m_2}{3} & 0 \\
0 & \frac{m_2}{6} & 0 & \frac{m_2}{3}
\end{bmatrix} (2)
\]

\[
\mathbf{J} = \begin{bmatrix}
2(x_1 - x_0) & 2(y_1 - y_0) & 0 & 0 \\
-2(x_2 - x_1) & -2(y_2 - y_1) & 2(x_2 - x_1) & 2(y_2 - y_1)
\end{bmatrix} (3)
\]
\[
\dot{\phi} = \begin{bmatrix}
0 \\
\frac{2}{m_1 + m_2 g} \\
0 \\
-\frac{m_2 g}{2}
\end{bmatrix}
\]  

(4)

The kinematic relations between the natural and relative coordinates are given below:

\[
\begin{align*}
\mathbf{v}_1 &= \begin{bmatrix} \dot{x}_1 \\ \dot{y}_1 \\ \dot{z}_1 \end{bmatrix} = \dot{c}_1 + \omega_1 \times r_1 = [-r_1 \times 1 \ 1]_N \omega_1 \\
\mathbf{v}_2 &= \begin{bmatrix} \dot{x}_2 \\ \dot{y}_2 \\ \dot{z}_2 \end{bmatrix} = \dot{c}_2 + \omega_2 \times r_2 = [-r_2 \times 1 \ 1]_N \omega_2
\end{align*}
\]  

(5a, 5b)

Then,

\[
\begin{bmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \end{bmatrix} = \begin{bmatrix} -r_1 \times 1 & 1 & 0 & 0 \\ 0 & 0 & -r_2 \times 1 & 1 \end{bmatrix} \begin{bmatrix} \omega_1 \\ \dot{c}_1 \\ \omega_2 \\ \dot{c}_2 \end{bmatrix}
\]  

(6)

where \(1\) and \(O\) are the \(3 \times 3\) identity and null matrix respectively.

The equation (6) can be written in compact form as given below:

\[
\mathbf{v} = R \mathbf{t}
\]

(7)

From [5], we have twist vector \(\mathbf{t} = N_l N_d \dot{\theta}\), therefore the above can be written as \(\mathbf{v} = R N_l N_d \dot{\theta}\), and in compact form \(\mathbf{v} = T \dot{\theta}\), where \(T = R N_l N_d\), which gives \(\dot{\mathbf{v}} = \ddot{T} \dot{\theta} + T \ddot{\theta}\). The velocity transformation matrix \(T\) for the two link system is given below:

\[
T = \begin{bmatrix}
-a_1 \sin \theta_1 & 0 \\
-\frac{a_1 \sin \theta_1 - a_2 \sin \theta_{12}}{a_1 \cos \theta_1 + a_2 \cos \theta_{12}} & \frac{a_1 \cos \theta_1 + a_2 \cos \theta_{12}}{a_1 \cos \theta_1 + a_2 \cos \theta_{12}} \\
\end{bmatrix}
\]

(8)

Put the value of the \(\ddot{\mathbf{v}}\) in eq. (1), and pre-multiply the transpose of \(T\) to the both side of eq. (1). This operation transforms the equation of motion in relative coordinates. The generalized form of equation of motion for the serial chain manipulator in relative coordinates is given below:

\[
I_r \ddot{\theta} + C \dot{\theta} = \mathbf{\tau}
\]

(9)

Where \((I_r)\) is the GIM in relative coordinates, \(C\) is the matrix of convective inertia (MCI) and \(\mathbf{\tau}\) is the generalized force vector (GFV). The transformed expression in relative coordinates for the GIM, MCI and GFV are given below:

\[
I_r = \begin{bmatrix}
\frac{3}{3} m_1 a_1^2 + \frac{3}{3} m_2 a_2^2 & m_2 a_2^2 \cos \theta_2 \\
\frac{3}{3} m_2 a_2^2 \cos \theta_2 & \frac{3}{3} m_2 a_2^2 \cos \theta_2 \\
\frac{3}{3} m_2 a_2^2 \cos \theta_2 & \frac{3}{3} m_2 a_2^2 \cos \theta_2 \\
\end{bmatrix}
\]

(10)

\[
C = \begin{bmatrix}
-\frac{1}{2} m_2 a_1 a_2 \sin \theta_2 \hat{\theta}_2 & \frac{1}{2} m_2 a_1 a_2 \sin \theta_2 \hat{\theta}_{12} \\
\frac{1}{2} m_2 a_1 a_2 \sin \theta_2 \hat{\theta}_1 & 0 \\
\end{bmatrix}
\]

(11)

\[
\mathbf{\tau} = \begin{bmatrix}
\tau_1 - \left(\frac{m_1 a_1}{2} + m_2 a_1\right) g \cos \theta_1 - \frac{m_2 a_2}{2} g \cos \theta_{12} \\
\tau_2 - \frac{m_2 a_2}{2} g \cos \theta_{12} \\
\end{bmatrix}
\]

(12)
2.2 Triangular Platform of 3-RRR Parallel Manipulator

The triangular platform of the manipulator is shown in Figure 2. This triangular platform is the part of the 3-RRR parallel manipulator and the nomenclature is as per the complete figure, will be shown later. The formulation for the velocity transformation matrix is given below:

$$
\begin{bmatrix}
\dot{x}_2 \\
\dot{y}_2 \\
\dot{z}_2 
\end{bmatrix} = \mathbf{c}_7 + \omega_7 \times \mathbf{e}_1 = [-\mathbf{e}_1 \times 1 \quad 1] \begin{bmatrix} \omega_7 \\ \mathbf{c}_7 \end{bmatrix} \quad (13a)
$$

Similarly, \( \mathbf{v}_4 \) and \( \mathbf{v}_6 \) can be written, which are given below:

$$
\begin{bmatrix}
\dot{x}_4 \\
\dot{y}_4 \\
\dot{z}_4 
\end{bmatrix} = \mathbf{c}_7 + \omega_7 \times \mathbf{e}_2 = [-\mathbf{e}_2 \times 1 \quad 1] \begin{bmatrix} \omega_7 \\ \mathbf{c}_7 \end{bmatrix} \quad (13b)
$$

$$
\begin{bmatrix}
\dot{x}_6 \\
\dot{y}_6 \\
\dot{z}_6 
\end{bmatrix} = \mathbf{c}_7 + \omega_7 \times \mathbf{e}_3 = [-\mathbf{e}_3 \times 1 \quad 1] \begin{bmatrix} \omega_7 \\ \mathbf{c}_7 \end{bmatrix} \quad (13c)
$$

The eq. (13) can be written in combined form as given below:

$$
\begin{bmatrix}
\mathbf{v}_2 \\
\mathbf{v}_4 \\
\mathbf{v}_6 
\end{bmatrix} = \begin{bmatrix}
-\mathbf{e}_1 \times 1 & 1 & 0 & 0 \\
-\mathbf{e}_2 \times 1 & 0 & 1 & 0 \\
-\mathbf{e}_3 \times 1 & 0 & 0 & 1 
\end{bmatrix} \begin{bmatrix} \omega_7 \\ \mathbf{c}_7 \end{bmatrix} \quad \text{OR } \mathbf{v} = \mathbf{R} \dot{\theta} \quad (14)
$$

By using the definition of DeNOC, the velocity transformation matrix \( \mathbf{T} \) can be written as

$$
\begin{bmatrix}
\mathbf{v}_2 \\
\mathbf{v}_4 \\
\mathbf{v}_6 
\end{bmatrix} = \begin{bmatrix}
-e_1 \sin(\theta_7 - 150^\circ) & 1 & 0 & 0 \\
e_1 \cos(\theta_7 - 150^\circ) & 0 & 1 & 0 \\
-e_2 \sin(\theta_7 - 30^\circ) & 1 & 0 & 0 \\
e_2 \cos(\theta_7 - 30^\circ) & 0 & 1 & 0 \\
-e_3 \sin(\theta_7 + 90^\circ) & 1 & 0 & 0 \\
e_3 \cos(\theta_7 + 90^\circ) & 0 & 1 & 0 
\end{bmatrix} \begin{bmatrix} \dot{x}_c \\
\dot{y}_c \\
\dot{z}_c \end{bmatrix} \quad \text{OR } \mathbf{v} = \mathbf{T} \dot{\theta} \quad (15)
$$

where \( \mathbf{T} \) is the velocity transformation matrix for the triangular platform.
3 FOUR-BAR MECHANISM

A four-bar mechanism is shown in Figure 3a, in which both the coordinate systems are used to define the configuration of the mechanism. The mechanism is divided into two sub-systems as shown in Figure 3b. First, the equation of motion is derived using the natural coordinates and then transformed to the relative coordinates using the velocity transformation matrix. The velocity transformation matrix for the four-bar mechanism is the combination of the transformation matrix for the one link and two link system respectively. The links length (m) and mass (Kg) are $l_1 = 0.038; l_2 = 0.1152; l_3 = 0.1152$ and $m_1 = 1.5; m_2 = 3; m_3 = 5$ respectively. The initial configuration of the mechanism is $x_1 = -0.038; y_1 = 0; x_2 = 0.0257; y_2 = 0.09595; \dot{x}_1 = 0; \dot{y}_1 = 0; \dot{x}_2 = 0; \dot{y}_2 = 0$. The results for the forward dynamics for the free-fall under gravity is shown in Figure 4a and 4b.

Figure 3a. Four-bar mechanism.  
Figure 3b. Four-bar mechanism with cut joint.

Figure 4a. Linear displacement $(x_1, y_1)$ variation with time.  
Figure 4b. Linear displacement $(x_2, y_2)$ variation with time.
The natural and relative coordinates are compared with respect to CPU time and constraint violation. The result for the constraint violation is shown in Figure 5a and 5b.

Figure 5a. Constraint violation at position and velocity level (natural coordinates).

Figure 5b. Constraint violation at position and velocity level (relative coordinates).

The Table 1. shows the numerical values of the CPU time and constraint violation.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Natural Coordinates</th>
<th>Relative Coordinates</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU Time (sec)</td>
<td>0.636068</td>
<td>0.772822</td>
</tr>
<tr>
<td>Constraint Violation at position level</td>
<td>5.264e-5</td>
<td>2.827e-4</td>
</tr>
<tr>
<td>Constraint violation at velocity level</td>
<td>3.991e-5</td>
<td>4.238e-4</td>
</tr>
</tbody>
</table>

4 3-RRR PARALLEL MANIPULATOR

The 3-RRR parallel manipulator is shown in Figure 6a. The manipulator is divided in four sub-systems, in which three sub-systems are of two link system and fourth sub-system is the triangular platform as shown in Figure 6b. The velocity transformation matrix for this manipulator is the combination of the transformation matrix for the two link system and the triangular platform as given in section 2. The links length (m) and mass (Kg) are \( l_1 = 0.4; \ l_2 = 0.6; \ l_3 = 0.4; \ l_4 = 0.6; \ l_5 = 0.6; \ l_6 = 0.4; \ l_7 = 0.4; \) and \( m_1 = 3; \ m_2 = 4; \ m_3 = 8; \ m_4 = 4; \ m_5 = 4; \ m_6 = 3; \ m_7 = 3 \) respectively.
Figure 6a. 3-RRR parallel manipulator.

$x_c = (x_2 + x_3 + x_4)/3$
$y_c = (y_2 + y_3 + y_4)/3$

Figure 6b. Four sub-systems of 3-RRR parallel manipulator.
The initial configuration of the mechanism is $x_1 = 0.2; y_1 = 0.3464; x_2 = 0.79088; y_2 = 0.4505; x_3 = 0.508; y_3 = 0.1676; x_4 = 0.8964; y_4 = 0.07; x_5 = 0.8; y_5 = -0.3464; x_6 = 0.8964; y_6 = 0.67; \dot{x}_1 = 0; \dot{y}_1 = 0; \dot{x}_2 = 0; \dot{y}_2 = 0; \dot{x}_3 = 0; \dot{y}_3 = 0; \dot{x}_4 = 0; \dot{y}_4 = 0; \dot{x}_5 = 0; \dot{y}_5 = 0; \dot{x}_6 = 0; \dot{y}_6 = 0$. The result for the forward dynamics for the $(x_1, y_1)$ coordinate, free-fall under gravity, is shown in Figure 7. This result is validated with the in-house developed software ReDySim.

![Figure 7](image)

**Figure 7.** Validation of the result for $(x_1, y_1)$ with ReDySim.

The results for the constraint violation and energy drift for both the coordinate systems are shown in Figure 8a and 8b.

![Figure 8a](image)

**Figure 8a.** Results for constraints violation and energy drift (natural coordinates).

![Figure 8b](image)

**Figure 8b.** Results for constraints violation and energy drift (relative coordinates).
The Table 2. shows the numerical values of the CPU time, constraint violation, and energy drift.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Natural Coordinates</th>
<th>Relative Coordinates</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU Time (sec)</td>
<td>0.613999</td>
<td>0.617357</td>
</tr>
<tr>
<td>Constraint Violation at position level</td>
<td>4.355e-5</td>
<td>6.617e-3</td>
</tr>
<tr>
<td>Constraint violation at velocity level</td>
<td>3.41e-5</td>
<td>2.62e-5</td>
</tr>
<tr>
<td>Energy drift (Joule)</td>
<td>1.591e-3</td>
<td>6.22e-4</td>
</tr>
</tbody>
</table>

5 RESULTS AND DISCUSSION
The results for the four-bar and 3-RRR parallel manipulator have been tabulated for two different coordinate systems. It is worth noting that natural coordinates should be beneficial for the closed-loop MBS. Table 1. shows that the modelling of the four-bar mechanism using natural coordinates is better in comparison of the relative coordinates in all aspects, i.e., constraint violation at position as well as at velocity level and in terms of CPU time. As we have mentioned that these coordinates do not have any sine and cosine terms. Therefore, the results should be better in comparison to the relative coordinates as reveals by Table 1.

But in case of the 3-RRR parallel manipulator, degree-of-freedom (DOF) of the mechanism is three. Therefore, the size of the GIM using natural coordinate is large as compared to relative coordinates. The CPU time in case of natural coordinates is not significantly small. It means, the large DOF systems will require more CPU time using natural coordinates as compared to the relative coordinates. However, the constraint violation at position level is lesser in natural coordinate as compared to the relative coordinates. Constraint violation at velocity level, the difference between the two coordinate system is not significant. The comparison with respect to the energy drift is significant and yields towards relative coordinates.

6 CONCLUSION
From Table 2. we can conclude that the use of natural coordinates for the dynamic analyses is inefficient as the DOF of the system increases. But, the formulation of the equations of motion is straight forward. The results using the natural coordinates are more accurate than relative coordinates. Therefore the combination of the two coordinate system for dynamic formulation can be efficient and accurate.

REFERENCES


The accuracy of integrated rotational quaternion and angular velocity from curvature

Anita Treven, Miran Saje

Faculty of Civil and Geodetic Engineering
University of Ljubljana
Jamova cesta 2, 1000 Ljubljana, Slovenia
[Anita.Treven, Miran.Saje]@fgg.uni-lj.si

ABSTRACT
The paper discusses the accuracy of the numerical integration of the rotational quaternion and the angular velocity from a given time- and space-dependent curvature. An accurate and stable integration method is needed in the dynamic analysis of non-linear spatial beams, if the formulation is based on the curvature interpolation. The equations for the curvature and the angular velocity in terms of rotational quaternions are formally equal, and differ only in space- and time derivative. Several numerical methods for the integration of rotations from angular velocity have been proposed. The problem of integrating rotations from the angular velocity occurs in various fields such as navigation, robotics and computer graphics. However, a majority of these methods fail to preserve the unity constraint of the rotational quaternion, and experience a sudden instability due to the horizontal stretching of the response curves. So far an exact closed-form solution has only been possible for the angular velocity being constant in time (or the curvature constant in space). Two numerical examples are given in the paper, the first one considering planar rotations, for which the presented method of integration gives the exact solution, and the second one which considers space rotations. The accuracy of the integrated rotational quaternion and the angular velocity from a given curvature is assessed through comparisons of analytically and numerically obtained values. The effect of reducing the size of the space- and time computational steps is discussed.

Keywords: Curvature, Angular velocity, Rotational quaternion, Integration of rotational quaternion from curvature, Integration of angular velocity from curvature.

INTRODUCTION

The problem of integrating rotational quaternion and angular velocity from a known time- and space-dependent curvature is met in the dynamic analysis of non-linear spatial beams, if the numerical formulation is based on curvature, as in [1]. The rotational quaternion and the angular velocity are obtained by solving a homogeneous and a non-homogeneous system of quasi-linear differential equations, respectively. To avoid singularities and make the formulation more convenient, rotational quaternions are used for the parametrization of rotations. Curvature is a function of rotational quaternion and its space derivative. The angular velocity is formally the same as the curvature in terms of rotational quaternions, but time is used rather than the space as a variable. Integrating rotations from the curvature is therefore equivalent to integrating rotations from the angular velocity. The problem is met in various fields, such as navigation, robotics, computer graphics and optical tracking. Several numerical solution methods for the integration of rotations have already been proposed [2, 3, 4, 5]. However, an exact closed-form solution has only been possible for the angular velocity being constant in time (or the curvature constant in space). The majority of these methods also fail to preserve the unity constraint of the rotational quaternion, and experience a sudden instability due to the horizontal stretching of quaternion response curves. The aim of the paper is to outline the numerical method, which overcomes these disadvantages, and to assess its accuracy. The method is, in detail, presented in [6]. It enables the integration of the angular velocity from the curvature (and wise-versa). The identical rotational quaternion...
must be obtained by integrating either the given curvature or the related given angular velocity, which implies that the two quantities are not independent. The solutions are exact, if certain conditions regarding components of the curvature vector are satisfied. Two numerical examples are presented, the first one considering planar rotations, for which the present method of integration gives the exact solution, and the second one considering space rotations. The accuracy of the rotational quaternion and angular velocity obtained from a given curvature is assessed through the comparison of analytically and numerically derived values. The effect of reducing the size of the space-, time- or both computational steps is discussed. The present method is appropriate for use in the finite-element type of numerical formulations of the dynamics of spatial beams. Therefore it is desired that magnitudes of errors of integrated quantities are comparable. The maximum errors of numerically obtained rotational quaternions and angular velocity are normalized and their orders of magnitude compared.

2 INTEGRATION METHOD

2.1 Parametrization of rotations

A choice of the parametrization of rotations influences the stability of the integration method [4]. The use of the rotational vector with three independent components can lead to singularities. In order to avoid singularities and make the formulation of the problem more convenient, the rotational quaternion, whose four components are mutually dependent, is employed in parametrizing rotations. Some important properties of quaternions are listed below. For further details see, e.g. [7].

An arbitrary quaternion \( \hat{a} \) is composed from a scalar part, \( a_0 \), and a three-component vector part, \( \vec{a} \):

\[
\hat{a} = a_0 + \vec{a}.
\]  

(1)

In the quaternion algebra, several operations are defined: the summation, the multiplication by a scalar and the multiplication by a quaternion, herein denoted by \( \circ \). The latter is associative, but it is not commutative. A conjugated quaternion \( \hat{a}^\ast \) is defined as \( \hat{a}^\ast = a_0 - \vec{a} \). A quaternion, whose scalar part is equal to zero, is called the pure quaternion. Hence, it is possible to represent any three-component vector as a pure quaternion.

The rotational quaternion is defined with unity vector \( \vec{n} \), which represents the axis of the rotation, and with the angle of rotation \( \vartheta \), as shown in Eq. (2):

\[
\hat{q} = \cos \frac{\vartheta}{2} + \vec{n} \sin \frac{\vartheta}{2}.
\]  

(2)

It is easy to prove that the rotational quaternion is unitary by definition.

2.2 Curvature and angular velocity in terms of rotational quaternions

The relation between the curvature quaternion, \( \hat{\kappa} \), and the rotational quaternion in the quaternion body basis is shown in Eq. (3):

\[
\hat{\kappa} = 2\hat{q}^\ast \circ \dot{\hat{q}}.
\]  

(3)

Here \( \hat{q}^\ast \) is the conjugated rotational quaternion, and the prime (‘) denotes the space derivative. For a given curvature, Eq. (3) constitutes the homogeneous system of four differential equations with variable coefficients and, together with the initial condition \( \hat{q}_{0,0} = \hat{q}(x_1,t) \), forms an initial-value problem for \( \dot{\hat{q}} \).

The angular velocity quaternion in the quaternion body basis, \( \hat{\omega} \), is defined similarly, only that the space derivative of the rotational quaternion is replaced by the time derivative:

\[
\hat{\omega} = 2\hat{q}^\ast \circ \dot{\hat{q}}.
\]  

(4)
For a given angular velocity, Eq. (4) constitutes the homogeneous system of four differential equations with variable coefficients and, together with the initial condition \( \hat{\mathbf{q}}_{w} = \hat{\mathbf{q}}(x, t_1) \), forms an initial-value problem for \( \hat{\mathbf{q}} \). By integrating either curvature or angular velocity, belonging to the same point at the same time, the identical rotational quaternion must be obtained, which implies that the two quantities are not independent. Their mutual relation, which is obtained by the elimination of the rotational quaternion from Eqs. (3) and (4), reads:

\[
\dot{\mathbf{q}}' = \frac{1}{2}[\dot{\mathbf{q}} \times \ddot{\mathbf{k}} - \ddot{\mathbf{k}} \times \dot{\mathbf{q}}] + \dot{\mathbf{k}}.
\]

Here \( \dot{\mathbf{q}}' \) is the space derivative of the angular velocity, and \( \ddot{\mathbf{k}} \) is the time derivative of the curvature. The angular velocity and curvature quaternions are both pure quaternions, whose vector parts are identical to the angular velocity and curvature respectively in more classic vector form. Hence, Eq. (5) represents a system of three non-homogeneous differential equations with variable coefficients. When the curvature is a given function, Eq. (5) along with the initial condition \( \hat{\mathbf{q}}_{w} = \hat{\mathbf{q}}(x, t_1) \) constitutes an initial-value problem for \( \hat{\mathbf{q}} \). If the angular velocity and the initial condition \( \dot{\mathbf{k}}_{w} = \dot{\mathbf{k}}(x, t_1) \) are given, Eq. (5) represents an initial-value problem for \( \dot{\mathbf{k}} \).

### 2.3 Solution of the initial-value problem

Exact solutions for homogeneous and non-homogeneous systems of differential equations with variable coefficients can be found, if certain conditions are satisfied [8]. For convenience the component form of Eqs. (3) and (5) is used and they are rearranged into matrices having varying, yet known coefficients, and arrays of unknown functions. The solution for the rotational quaternion is presented in Eq. (6), where \( \mathbf{q} \) is an array of four unknown components of the rotational quaternion, and \( K_{4D} \) is a given 4x4 skew-symmetric matrix:

\[
\mathbf{q}(x, t) = e^{\int_{t_1}^{t} K_{4D} d\xi} \mathbf{q}(x_1, t) \quad \mathbf{q}_0 = \mathbf{q}(x_1, t).
\]

Since Eq. (5) represents a non-homogeneous differential equation, the resulting solution for the angular velocity is composed from general and particular solutions:

\[
\mathbf{\omega}(x, t) = e^{-\int_{t_1}^{t} K_{3D} d\eta} \mathbf{\omega}(x_1, t) + \int_{t_1}^{t} e^{-\int_{\eta}^{t} K_{3D} d\eta} K_{3D}(\eta, t) d\eta, \quad \mathbf{\omega}_0 = \mathbf{\omega}(x_1, t).
\]

Here \( \mathbf{\omega} \) is an array of three unknown components of the angular velocity, \( K_{3D} \) is a known 3x3 skew-symmetric matrix, and \( \dot{K}_{3D} \) is the time derivative of its axial vector. The solution for the case where the angular velocity is given can be written by analogy and is not discussed hereinafter. The condition for the solutions to be exact [8] can be rewritten as two simple scalar conditions, once we consider that the matrices \( K_{4D} \) and \( K_{3D} \) are skew-symmetric. Scalar components of the given curvature vector are denoted as \( \kappa_1, \kappa_2 \) and \( \kappa_3 \). The conditions now read:

\[
\kappa_1(x, t) \kappa_2'(x, t) = \kappa_2(x, t) \kappa_1'(x, t),
\]

\[
\kappa_1(x, t) \kappa_3'(x, t) = \kappa_3(x, t) \kappa_1'(x, t).
\]

In fact Eqs. (8) and (9) require that the components of the curvature be arbitrarily scaled functions of the same form in time and space. This requirement is fulfilled, e.g. for a constant curvature and also for planar rotations (where the axis of the rotation is constant, while the angle of the rotation is an arbitrary function of time and space).

It can also be proven that even if these conditions are not met, the proposed solutions approach the exact solutions when the observed interval \([x_1, x_2]\) decreases. Hence, Eqs. (6) and (7) represent approximate solutions, which converge toward the exact solutions in the limit.

The matrix exponential of a 3D or a 4D skew-symmetric matrix can be expressed by the exact closed-form Rodrigues formulas for 3D and 4D, respectively. The integrals in Eqs. (6) and (7) are
generally not analytically solvable, and even if they are, the numerical evaluation of the analytic solution can be computationally demanding. Here a 2-point Gaussian numerical integration is employed. This makes it possible for the proposed solution to be written in a convenient closed form.

3 NUMERICAL EXAMPLES

The validity of the present method was tested in [6], where it was shown that the integration gives very satisfactory results. Further accuracy tests are performed in the present paper. The main objective of the tests is to determine the effect of a decrease of time- and/or space computational steps and to compare orders of magnitudes of normalized global errors of the integrated rotational quaternion and angular velocity. In the first numerical example, planar rotations are studied so that the curvature identically satisfies the conditions for the solution to be exact. Fully general space rotations not tested in [6], define the curvature for the second numerical example; there the conditions of exactness are not fulfilled.

In both examples, the rotational quaternion is given as an analytical function of time and space, described by a chosen angle and the axis of rotation, in accordance with Eq. (2). The analytical curvature and angular velocity are then obtained through Eqs. (3) and (4). Eqs. (6) and (7) are employed so that the numerical rotational quaternion and the angular velocity are computed and compared with the analytical ones. Since there is no integration or derivation with respect to time in Eq. (6), \( \tau \) is chosen to be constant with value 5 and \( x \) runs between 0 and 10 when the integration of rotational quaternion is performed. By constrast the angular velocity is computed on the domain \( \{(x,t) : 0 \leq x \leq 10; 0 \leq t \leq 5\} \), which is due to the presence of the derivation with respect to time in one of the integrals in Eq. (7). Small computational steps \( \Delta x = x_2 - x_1 \) and \( \Delta t = t_2 - t_1 \) are used. Maximum global errors in components and in the norm of both quantities are normalized (i.e. divided by the maximum value of the corresponding analytically obtained norm).

3.1 Planar rotations

In the first example, the angle and the axis of rotation are chosen to be \( \vartheta = \frac{x \tau}{100} \) and \( \mathbf{n} = \{0.8, -0.2, \sqrt{0.32}\} \). The axis of the rotation is constant, which results in planar rotations. The rotational quaternion, given in Eq. (10), defines the curvature and angular velocity vectors as written in Eqs. (11) and (12):

\[
q(x,t) = \begin{bmatrix}
\cos\left(\frac{x \tau}{100}\right) \\
0.8 \sin\left(\frac{x \tau}{100}\right) \\
-0.2 \sin\left(\frac{x \tau}{100}\right) \\
\sqrt{0.32} \sin\left(\frac{x \tau}{100}\right)
\end{bmatrix}, \quad (10)
\]

\[
k_{3D}(x,t) = \begin{bmatrix}
4\left(\frac{x \tau}{125}\right) \\
-1\left(\frac{x \tau}{125}\right) \\
2\sqrt{2}\left(\frac{x \tau}{125}\right)
\end{bmatrix}, \quad (11)
\]

\[
\omega_{3D}(x,t) = \begin{bmatrix}
6\left(\frac{x \tau}{125}\right) \\
-1.5\left(\frac{x \tau}{125}\right) \\
3\sqrt{2}\left(\frac{x \tau}{125}\right)
\end{bmatrix}. \quad (12)
\]

It is clear that the components of the curvature are differently scaled functions with the same time and space variation, which can also be observed from Fig. 1. The conditions in Eqs. (8) and (9) are therefore fulfilled and Eqs. (6) and (7) result in exact solutions for the rotational quaternion and the angular velocity. The maximum absolute errors in the norm and components of the rotational quaternion at \( t = 5 \) are presented in Tab. 1. As the rotational quaternion is unitary, these absolute errors are also the normalized errors. The numerical and analytical solutions match within the machine precision for all components and the norm of the rotational quaternion, and for all
computational steps ∆x. As the length of the steps decreases, the round-off errors increase slightly, however, which is due to a larger number of steps.

Figure 1: Planar rotations: Analytically obtained components of the curvature (left) and the angular velocity (right) on the domain \( \{(x,t) : 0 \leq x \leq 10; 0 \leq t \leq 5 \} \).

Table 1: Planar rotations: Maximum global absolute errors of components and norms of integrated rotational quaternion at \( t = 5 \).

<table>
<thead>
<tr>
<th>∆x</th>
<th>\text{err}(q_0)</th>
<th>\text{err}(q_1)</th>
<th>\text{err}(q_2)</th>
<th>\text{err}(q_3)</th>
<th>\text{err}(|q|)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5.5 \cdot 10^{-16}</td>
<td>3.9 \cdot 10^{-16}</td>
<td>1.7 \cdot 10^{-16}</td>
<td>3.9 \cdot 10^{-16}</td>
<td>3.3 \cdot 10^{-16}</td>
</tr>
<tr>
<td>0.2</td>
<td>1.3 \cdot 10^{-14}</td>
<td>6.4 \cdot 10^{-15}</td>
<td>1.9 \cdot 10^{-15}</td>
<td>5.1 \cdot 10^{-15}</td>
<td>6.6 \cdot 10^{-16}</td>
</tr>
<tr>
<td>0.1</td>
<td>1.3 \cdot 10^{-14}</td>
<td>1.0 \cdot 10^{-14}</td>
<td>2.4 \cdot 10^{-15}</td>
<td>7.4 \cdot 10^{-15}</td>
<td>1.4 \cdot 10^{-15}</td>
</tr>
<tr>
<td>0.01</td>
<td>1.8 \cdot 10^{-14}</td>
<td>1.3 \cdot 10^{-14}</td>
<td>3.8 \cdot 10^{-15}</td>
<td>1.0 \cdot 10^{-14}</td>
<td>1.9 \cdot 10^{-15}</td>
</tr>
<tr>
<td>0.001</td>
<td>1.8 \cdot 10^{-14}</td>
<td>1.5 \cdot 10^{-14}</td>
<td>4.3 \cdot 10^{-15}</td>
<td>9.8 \cdot 10^{-15}</td>
<td>4.2 \cdot 10^{-15}</td>
</tr>
</tbody>
</table>
The variation of analytically and numerically obtained components of the rotational quaternion over the interval \(\{0 \leq x \leq 10\}\) is shown in Fig. 2. Some differences between the numerical solution for \(\Delta x\) and the analytical solution can be noted, due to the fact, that the numerical solutions are only known at discrete points, and that the straight lines are drawn in between.

Absolute and normalized maximum global errors of components of the angular velocity, obtained via Eq. (7), are presented in Tab. 2. The absolute maximum differences between analytical and numerical solutions reach the order of magnitude \(10^{-13}\). These are divided by 150, which is the maximum norm of the analytically computed angular velocity. Thus the obtained normalized errors are denoted as \(err_{\text{norm}}(\omega_i), i = 1, 2, 3\). Their order of magnitude is \(10^{-16}\), which is again the order of machine round-off errors. Only two different combinations of sizes of computational steps are considered, as the solutions (6) and (7) are exact (within the machine precision) for any size of the computational step for this example, and the beneficial effect of decreasing length of computational steps cannot be observed.

Table 2: Planar rotations: Maximum global errors and maximum normalized global errors of components of integrated angular velocity.

<table>
<thead>
<tr>
<th>(\Delta x)</th>
<th>(\Delta t)</th>
<th>(err(\omega_1))</th>
<th>(err_{\text{norm}}(\omega_1))</th>
<th>(err(\omega_2))</th>
<th>(err_{\text{norm}}(\omega_2))</th>
<th>(err(\omega_3))</th>
<th>(err_{\text{norm}}(\omega_3))</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2</td>
<td>0.2</td>
<td>(5.7 \cdot 10^{-14})</td>
<td>(3.8 \cdot 10^{-16})</td>
<td>(7.5 \cdot 10^{-14})</td>
<td>(5.0 \cdot 10^{-16})</td>
<td>(5.7 \cdot 10^{-14})</td>
<td>(3.8 \cdot 10^{-16})</td>
</tr>
<tr>
<td>0.1</td>
<td>0.1</td>
<td>(1.3 \cdot 10^{-13})</td>
<td>(8.5 \cdot 10^{-16})</td>
<td>(9.6 \cdot 10^{-14})</td>
<td>(6.4 \cdot 10^{-16})</td>
<td>(1.1 \cdot 10^{-13})</td>
<td>(7.6 \cdot 10^{-16})</td>
</tr>
</tbody>
</table>

Figure 2: Planar rotations: Variation of the components of the numerically and analytically obtained rotational quaternion over the interval \(\{0 \leq x \leq 10\}\) for \(t = 5, \Delta x = 0.1, 0.01\).
3.2 Space rotations

Space rotations are defined with a spatially varying, yet unitary axis of rotation \( \vec{n} = \{ \cos(0.25x), \sqrt{0.3}\sin(0.25x), \sqrt{0.7}\sin(0.25x) \} \), and a time and space dependent angle of rotation \( \vartheta = 2x^2 \sin(t) \). The related rotational quaternion reads:

\[
q(x,t) = \begin{bmatrix}
\cos(x^2 \sin(t)) \\
\cos(0.25x) \sin(x^2 \sin(t)) \\
\sqrt{0.3}\sin(0.25x) \sin(x^2 \sin(t)) \\
\sqrt{0.7}\sin(0.25x) \sin(x^2 \sin(t)) 
\end{bmatrix}
\] (13)

Due to the symmetrical form of Eq. (5) one may expect, that the components of the corresponding curvature and angular velocity vectors have symmetrical forms. However, Eqs. (14) and (15) do not confirm the expectations. The time and space variation of the components of the analytically obtained curvature and angular velocity are shown in Fig. 3.

Figure 3: Space rotations: Analytically obtained components of the curvature (left) and the angular velocity (right) on the domain \( \{(x, t) : 0 \leq x \leq 10; 0 \leq t \leq 5\} \).
\[
\mathbf{k}_{3D} = \begin{bmatrix}
\frac{\sqrt{70}}{20} (1 - \cos^2(x^2 \sin(t))) + \frac{\sqrt{70}}{20} (\cos\left(\frac{x}{4}\right) \cos(x^2 \sin(t)) \sin(x^2 \sin(t)) + 8 \sin\left(\frac{x}{4}\right) \sin(t)) \\
-\frac{\sqrt{30}}{20} (1 - \cos^2(x^2 \sin(t))) + \frac{\sqrt{30}}{20} (\cos\left(\frac{x}{4}\right) \cos(x^2 \sin(t)) \sin(x^2 \sin(t)) + 8 \sin\left(\frac{x}{4}\right) \sin(t)) \\
4 \cos\left(\frac{x}{4}\right) \sin(t) - \frac{1}{2} \sin\left(\frac{x}{4}\right) \cos(x^2 \sin(t)) \sin(x^2 \sin(t))
\end{bmatrix}
\]

(14)

\[
\omega_{3D} = \begin{bmatrix}
2x^2 \cos\left(\frac{x}{4}\right) \cos(t) \\
\frac{\sqrt{70}}{20} x^2 \sin\left(\frac{x}{4}\right) \cos(t) \\
\frac{\sqrt{30}}{20} x^2 \sin\left(\frac{x}{4}\right) \cos(t)
\end{bmatrix}
\]

(15)

Note that the difference in the form of the components of the both quantities, although not so obvious, can also be observed in the previous example of planar rotations. Eq. (14) clearly shows that the conditions (8) and (9) for the solutions to be exact are not fulfilled. Hence, the decrease of the computational step should improve the accuracy of the integration method. The following computational steps \(\Delta x\) were used for the integration of the rotational quaternion from the curvature: 1, 0.2, 0.1, 0.01 and 0.001.

![Figure 4: Space rotations: Variation of the components of the numerically and analytically obtained rotational quaternion over the interval \(\{0 \leq x \leq 10\}\) for \(t = 5, \Delta x = 0.1, 0.01\).](image)

The maximum absolute errors in the norm and components of the rotational quaternion at \(t = 5\) are presented in Tab. 3. As the rotational quaternion is unitary, these absolute errors are equal to the normalized errors. The errors in the components are quite big for \(\Delta x = 1\), while for \(\Delta x = 0.01\) the results match within a graphical precision, as it is shown in Fig. 4. A roughly quadratic convergence can be observed. The unit norm of the rotational quaternion is always preserved.
within the machine precision for any length of the computational step. From Fig. 4 it is also
obvious, that no horizontal stretching takes place in quaternion response curves.

Table 3: Space rotations: Maximum global errors of components and norms of integrated rota-
tional quaternion at \( t = 5 \).

<table>
<thead>
<tr>
<th>( \Delta t )</th>
<th>( \text{err}(\omega_0) )</th>
<th>( \text{err}(\omega_{1\text{norm}}) )</th>
<th>( \text{err}(\omega_2) )</th>
<th>( \text{err}(\omega_{2\text{norm}}) )</th>
<th>( \text{err}(|\mathbf{\ell}|) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.1 \cdot 10^{-1}</td>
<td>9.7 \cdot 10^{-1}</td>
<td>4.4 \cdot 10^{-1}</td>
<td>4.5 \cdot 10^{-1}</td>
<td>2.2 \cdot 10^{-16}</td>
</tr>
<tr>
<td>0.2</td>
<td>3.4 \cdot 10^{-1}</td>
<td>7.6 \cdot 10^{-1}</td>
<td>6.2 \cdot 10^{-1}</td>
<td>5.0 \cdot 10^{-1}</td>
<td>5.5 \cdot 10^{-16}</td>
</tr>
<tr>
<td>0.1</td>
<td>7.0 \cdot 10^{-2}</td>
<td>3.9 \cdot 10^{-1}</td>
<td>3.4 \cdot 10^{-1}</td>
<td>2.8 \cdot 10^{-1}</td>
<td>6.6 \cdot 10^{-16}</td>
</tr>
<tr>
<td>0.01</td>
<td>2.9 \cdot 10^{-5}</td>
<td>3.1 \cdot 10^{-3}</td>
<td>4.6 \cdot 10^{-3}</td>
<td>4.4 \cdot 10^{-3}</td>
<td>4.0 \cdot 10^{-15}</td>
</tr>
<tr>
<td>0.001</td>
<td>2.7 \cdot 10^{-7}</td>
<td>3.1 \cdot 10^{-5}</td>
<td>4.6 \cdot 10^{-5}</td>
<td>4.4 \cdot 10^{-5}</td>
<td>6.2 \cdot 10^{-15}</td>
</tr>
</tbody>
</table>

Absolute and normalized maximum global errors of components of the angular velocity, when
obtained from Eq. (7), are presented in Tab. 4. The absolute maximum differences between ana-
tycial and numerical solutions are divided by 200, which is the maximum norm of the analytically
computed angular velocity, to obtain the normalized errors. Several combinations of the space
and time computational steps were used: i) \( \Delta x = \Delta t = 0.2 \), ii) \( \Delta x = \Delta t = 0.1 \), iii) \( \Delta x = 0.2 \) and
\( \Delta t = 0.1 \), iv) \( \Delta x = 0.1 \) and \( \Delta t = 0.2 \) and v) \( \Delta x = 0.01 \) and \( \Delta t = 0.2 \).

Table 4: Space rotations: Maximum global errors and maximum normalized global errors of com-
ponents of integrated angular velocity.

<table>
<thead>
<tr>
<th>( \Delta x )</th>
<th>( \Delta t )</th>
<th>( \text{err}(\omega_0) )</th>
<th>( \text{err}(\omega_{1\text{norm}}) )</th>
<th>( \text{err}(\omega_2) )</th>
<th>( \text{err}(\omega_{2\text{norm}}) )</th>
<th>( \text{err}(|\mathbf{\ell}|) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>i)</td>
<td>0.2</td>
<td>0.2</td>
<td>65.84</td>
<td>3.3 \cdot 10^{-1}</td>
<td>85.22</td>
<td>4.3 \cdot 10^{-1}</td>
</tr>
<tr>
<td>ii)</td>
<td>0.1</td>
<td>0.1</td>
<td>28.90</td>
<td>1.4 \cdot 10^{-1}</td>
<td>35.71</td>
<td>1.8 \cdot 10^{-1}</td>
</tr>
<tr>
<td>iii)</td>
<td>0.2</td>
<td>0.1</td>
<td>68.01</td>
<td>3.4 \cdot 10^{-1}</td>
<td>85.22</td>
<td>4.3 \cdot 10^{-1}</td>
</tr>
<tr>
<td>iv)</td>
<td>0.1</td>
<td>0.2</td>
<td>26.88</td>
<td>1.3 \cdot 10^{-1}</td>
<td>35.53</td>
<td>1.8 \cdot 10^{-1}</td>
</tr>
<tr>
<td>v)</td>
<td>0.01</td>
<td>0.2</td>
<td>0.40</td>
<td>2.0 \cdot 10^{-3}</td>
<td>0.47</td>
<td>2.3 \cdot 10^{-3}</td>
</tr>
</tbody>
</table>

From the results of the combinations i)–iv) it can be seen that decreasing \( \Delta t \) does not decrease
absolute errors. In contrast, decreasing \( \Delta x \) does. This happens due to the fact that all integrations
in Eq. (7) are conducted with respect to \( x \) and that the time derivative of the curvature is analytically
calculated, since the curvature is a given analytical function. A decrease of \( \Delta t \) would influence the
results if the curvature would be a function, interpolated over the values known only at \( \Delta t \)'s. Hence,
for the combination v), the biggest \( \Delta t \) is maintained and \( \Delta x \) is decreased further. The comparison
of the normalized errors of components of the integrated rotational quaternion and angular velocity
from Tabs. 3 and 4, respectively, shows a comparable order of magnitude for the same step \( \Delta x \).

The numerically obtained angular velocity, for both iv) and v) combinations of the computational
steps, is presented in Fig. 5 over the interval \( \{0 \leq x \leq 10\} \) for \( t = 1, 2, 3, 4, 5 \). An analytical solution
is added for the sake of comparison. The combination v) and the analytical values match perfectly
within the graphic precision.
Figure 5: Space rotations: Variation of the components of the numerically and analytically obtained angular velocity over the interval \( \{0 \leq x \leq 10\} \) for \( t = 1, 2, 3, 4, 5 \).

a) \( \Delta x = 0.1, \ \Delta t = 0.2 \)

b) \( \Delta x = 0.01, \ \Delta t = 0.2 \)
4 CONCLUSIONS

In concluding we wish to stress that the present method preserves unity constraint of the rotational quaternion and, in some specific cases (e.g. planar rotations), gives exact solutions. The method provides solutions that well converge towards the exact ones with the decrease of the space computational step size. The orders of magnitude of the maximum global errors in components of the integrated rotational quaternion and the related angular velocity are also alike. Finally, no horizontal stretching is observed in quaternion response curves, which is of a paramount importance for stability of integration.

REFERENCES


The Logarithmic Complexity Procedure for Parallel Multibody Dynamics Solution

Ladislav Mráz*, Michael Valášek *

* Faculty of Mechanical Engineering
Czech Technical University in Prague
Technická 4, Praha 6
166 07, Czech Republic
[ladislav.mraz, michael.valasek]@fs.cvut.cz

ABSTRACT

The paper presents the logarithmic complexity method for the solution of the multibody dynamics. The core of the method is the systematic reduction of the system size using the successive elimination of the solved variables. The resulting equations of motion are obtained using the Modified State Space method with application of the natural coordinates. As the consequence, the system of linear equations, which have to be solved, is symmetric positive definite and has a band structure. The whole system of equations is further divided into the smaller interconnected subsystems. The solved variables are systematically eliminated from the subsystems and thus the size of the problem being solved is reduced in each elimination step by factor 2. In the end, the root level is obtained with one subsystem only, which is solved. The rest of subsystems is further solved by back-substitution. This method can be combined with Cholesky decomposition and thus the greater effectivity is obtained.

Keywords: Elimination, Cholesky decomposition, Modified State Space, Parallelization

1 INTRODUCTION

The modeling of the multibody systems represents still growing demands on the computing power. The multibody systems being modeled are becoming greater, involving more bodies, more particles, more elements. Therefore, it is necessary to develop the new procedures for effective solution of such systems. The procedures have to correspond to the hardware being available nowadays. It means in particular the ability of parallel computation.

Many effective formalisms have been developed for the multibody dynamics solution. However, they parallel implementation is often limited. As an example the Composite Rigid Body (CRB) can be stated [16], or its modification for the flexible systems (CFB) [4]. These recursive methods perform good efficiency of $O(n^2)$ complexity, however, their recursivity disable the parallel realization. Similar situations hold for the effective Articulated Body Inertia method [16] with complexity $O(n)$.

Nowadays the parallel computation of the multibody dynamics is often required. Therefore, the completely other approaches have to be used. One promising procedure is the domain decomposition method [11-15]. It is very suitable for the finite element analysis or for the contact problems parallel solution.

For the parallel solution of the multibody systems with standard joints, the Divide and Conquer (DAC) method [1-3] is very powerful. The procedure with the logarithmic complexity $O(\log(n))$ is based on the systematic binary coupling and decoupling of the system elements. The applications show good performance and its logarithmic complexity seems to be the best for the parallel solvers of the multibody dynamics for the cases of the open kinematic chains.

The authors have tried to develop an algorithm solving the multibody dynamics in parallel, which would have the complexity $O(I)$. The developed procedures have been based on the decoupling of the system using the flexible joints and the heterogeneous integration [8,9], or the using of the iterative solution of the equations of motion (EOM) by leveraging of the traditional Jacobi iteration scheme or the artificial dynamics [10].
The research in this field has shoved, however, the presence of the dependence on the system size with resulting polynomial complexity $O(n^3)$ or the problematic setup of the algorithms for the large systems. Therefore, a new procedure similar to DAC have been developed, which can be even more powerful in the parallel multibody dynamics solution and it is the main topic of this article. The method using the natural coordinates for the multibody system description leverages the Schur complement technique for the EOM assembling. The resulting EOM system has the symmetric, positive definite and band structure, which is suitable for parallel solution. The smaller blocks can be found, into which the system can be decoupled. The particular blocks are in connection always with the neighbors only and therefore the procedure of systematic elimination can be designed, which has the logarithmic complexity. It has been shown, that the procedure is suitable for larger multibody systems (involving more than approx. 1000 bodies). For the smaller systems the Cholesky decomposition is more suitable. However, the logarithmic elimination can be used in connection with the Cholesky decomposition and thus to reach the better effectivity than the standard DAC algorithm.

2 THE LOGARITHMIC ELIMINATION METHOD

The assembling of the EOM system is necessary prior to the elimination procedure is applied. The Modified State Space method [5, 6] is adopted for this purpose with suitable choice of the natural coordinates. Using the Schur complement technique the EOM system is obtained with advantageous structure described further in this section.

2.1 Modified State Space with the natural coordinates

The mechanical system being modeled (Fig. 1) is a planar open kinematic chain with rotational joints. The reason for modeling such a system is the simplicity. It is, however, possible to apply following procedure to more complex spatial systems with general kinematic joints.

![Figure 1. Kinematic chain with rotational joints](image)

Each body in the chain is described by the set of redundant natural coordinates (Fig. 2). Its position is determined using the Cartesian coordinates of the centre of mass $S_i = [x_{Si}, y_{Si}]$ and the orientation of the body is represented by the unit vector $e_i = [x_{ei}, y_{ei}]$. Other bodies can be attached to this body in points $H_i$ and $H_{i+1}$.

![Figure 2. Body described by natural coordinates](image)
The coordinates \( s_i = [x_{Si}, y_{Si}, x_{e_i}, y_{e_i}]^T \) are defined for each body and they are arranged into the global coordinates-vector \( s = [s_1, s_2, ..., s_n]^T \).

The constraint equations of the redundant coordinates used for the description of the planar system (Fig. 1) can be easily derived [10] and they express the unit magnitudes of the vectors \( e_i \) and the relations between the centres of masses of the neighbouring bodies. Their general structure is:

\[
f(s) = 0
\]  

The time differentiation of the constraints gives

\[
f(s)\dot{s} = 0
\]  

where the matrix \( J \) is the Jacobian matrix with sparse block structure.

The modified momentum is introduced:

\[
p^* = p + J^T\mu
\]  

The symbol \( p \) denotes the classical momentum:

\[
p = M\dot{s}
\]  

and \( \mu \) is the vector of the new Lagrange multipliers. These new multipliers are related to the traditional ones as:

\[
\dot{\mu} = -\lambda
\]  

The mass matrix \( M_i \) of body \( i \) has the diagonal structure and it is constant consisting of the particular body masses and inertias: \( M_i = diag[m_i, m_i, I_i, I_i] \). The mass matrix of the whole kinematic chain has therefore also the diagonal structure:

\[
M = diag[M_1, M_2, ..., M_n]
\]

After substituting (4) into (3) it holds:

\[
p^* = M\dot{s} + J^T\mu
\]  

If the natural coordinates are used, following equation for the time derivatives of the modified momenta is obtained from the Lagrange equations of mixed type:

\[
\dot{p}^* = Q(s, \dot{s}) + J^T\mu
\]  

where \( Q \) is the vector of applied forces.

The equation (2) is stabilized using the Baumgarte stabilization and it is solved together with the equation (6):

\[
\begin{bmatrix} M & J^T \\ J & 0 \end{bmatrix} \begin{bmatrix} \dot{s} \\ \mu \end{bmatrix} = \begin{bmatrix} p^* \\ -\alpha f \end{bmatrix}
\]  

where \( \alpha \) is a positive parameter of stabilization.

In order to exploit the block structure of the Jacobian matrix and the diagonal structure of the mass matrix, the Schur complement method [10] has been adopted. The modified state space method is used for the assembling of EOM and the key equation which has to be solved is (8).

Instead of direct solution of the system, the involved equations are written down:

\[
M\dot{s} + J^T\mu = p^*
\]  

\[
J\dot{s} = -\alpha f(s)
\]  

From (9) the velocities \( \dot{s} \) can be expressed:

\[
\dot{s} = M^{-1}(p^* - J^T\mu)
\]  

and substituting (11) into (10) following expression is obtained:
\[ JM^{-1}(p^* - f^T \mu) = -\alpha f(s) \]  

(12)

After formal modification, the final equation for unknown \( \mu \) arises:

\[ JM^{-1}f^T \mu = \alpha f(s) + JM^{-1}p^* \]  

(13)

If the expressions are denoted:

\[ A = JM^{-1}f^T \quad b = \alpha f(s) + JM^{-1}p^* \]

the equation (13) can be rewritten in the form:

\[ A\mu = b \]  

(14)

This system of linear algebraic equations has symmetric positive definite system matrix \( A \) with band structure (Fig. 3).

After (14) is solved, the vector of new Lagrange multipliers \( \mu \) is obtained and the vector of velocities \( s \) is computed in (11). Afterwards, the time derivative of the modified momenta \( \dot{p}^* \) is computed (7) and the state derivatives \( \dot{s} \) and \( \dot{p}^* \) are numerically integrated in time, in order to get the new state of the system and the procedure is repeated.

2.2 The elimination procedure

The system (14) is the key one for the dynamics solution, since it is the linear system of interconnected equations. Following equations (11) for the velocity computation and (7) for the modified momentum derivatives are then easily parallel evaluated. Therefore, the attention is given to the system (14) only in this section. As stated before, the system has a sparse, band structure and there can be defined small blocks (Fig. 4), which represent the particular bodies in the kinematic chain.

![Figure 3. The structure of the matrix A](image)

![Figure 4. The blocks in the equations system](image)
Obviously, the system (14) can be naturally disassembled into the subsystems:

\[
\begin{align*}
A_{11}\mu_1 + A_{12}\mu_2 &= b_1 \\
A_{12}^T\mu_1 + A_{22}\mu_2 + A_{23}\mu_3 &= b_2 \\
A_{23}^T\mu_2 + A_{33}\mu_3 + A_{34}\mu_4 &= b_3 \\
&\vdots \\
A_{n-2,n-1}^T\mu_{n-2} + A_{n-1,n-1}\mu_{n-1} + A_{n-1,n}\mu_n &= b_{n-1} \\
A_{n-1,n}\mu_{n-1} + A_{nn}\mu_n &= b_n
\end{align*}
\] (15)

These subsystems are naturally composed by 3 scalar equations corresponding to the constraints of particular body. However, the size can be expanded as discussed further.

The particular unknown vectors \(\mu_i\) can be expressed from the even subsystems:

\[
\begin{align*}
\mu_2 &= A_{22}^{-1}(b_2 - A_{12}^T\mu_1 - A_{23}\mu_3) \\
\mu_4 &= A_{44}^T(b_4 - A_{34}\mu_3 - A_{45}\mu_5)
\end{align*}
\] (16)

These expressions can be further substituted into the odd equations (15) and thus the modified system is obtained:

\[
\begin{align*}
A_{11}\mu_1 + A_{12}A_{22}^{-1}(b_2 - A_{12}^T\mu_1 - A_{23}\mu_3) &= b_1 \\
A_{23}A_{22}^{-1}(b_2 - A_{12}^T\mu_1 - A_{23}\mu_3) + A_{33}\mu_3 + A_{34}A_{44}^{-1}(b_4 - A_{34}\mu_3 - A_{45}\mu_5) &= b_3 \\
&\vdots
\end{align*}
\] (17)

The compact form of the system (17) is:

\[
\begin{align*}
(A_{11} - A_{12}A_{22}^{-1}A_{12}^T)\mu_1 - A_{12}A_{22}^{-1}A_{23}\mu_3 &= b_1 - A_{12}A_{22}^{-1}b_2 \\
-A_{23}A_{22}^{-1}A_{12}\mu_1 + (A_{33} - A_{23}A_{22}^{-1}A_{23} - A_{34}A_{44}^{-1}A_{34})\mu_3 - A_{34}A_{44}^{-1}A_{45}\mu_5 &= b_3 - A_{23}A_{22}^{-1}b_2 - A_{34}A_{44}^{-1}b_4 \\
&\vdots
\end{align*}
\] (18)

After formal substitution the system can be written:

\[
\begin{align*}
1A_{11}\mu_1 + 1A_{12}\mu_3 &= 1b_1 \\
1A_{12}^T\mu_1 + 1A_{22}\mu_3 + 1A_{23}\mu_5 &= 1b_2 \\
&\vdots
\end{align*}
\] (19)

The mentioned substitutions are:

\[
\begin{align*}
(A_{11} - A_{12}A_{22}^{-1}A_{12}^T) &= 1A_{11} \\
-A_{12}A_{22}^{-1}A_{23} &= 1A_{12} \\
(A_{33} - A_{23}A_{22}^{-1}A_{23} - A_{34}A_{44}^{-1}A_{34}) &= 1A_{22} \\
-A_{34}A_{44}^{-1}A_{45} &= 1A_{23} \\
b_1 - A_{12}A_{22}^{-1}b_2 &= 1b_1 \\
b_3 - A_{23}A_{22}^{-1}b_2 - A_{34}A_{44}^{-1}b_4 &= 1b_2 \\
&\vdots
\end{align*}
\] (20)

It is clearly seen, that the system (19) is formally the same as (15). It has, however, a half number of items, since there are not present the even vectors \(\mu_2, \mu_4, \ldots\) anymore. The whole procedure can be repeated. Generally this elimination procedure can be written as follows: The particular subsystems are:

\[
A_{i-1,i}\mu_{i-1} + A_{i,i}\mu_i + A_{i,i+1}\mu_{i+1} = b_i
\] (21)

The particular vectors \(\mu_i\) expressed from (21) are:

\[
\mu_i = A_{ii}^{-1}(b_i - A_{i-1,i}\mu_{i-1} - A_{i,i+1}\mu_{i+1})
\] (22)

Evaluating this expressions for \(i=j-l\) and \(i=j+l\) and substituting into \(i=j\) yields:

\[
1A_{j-2,j}\mu_{j-2} + 1A_{j,j}\mu_{j} + 1A_{j,j+2}\mu_{j+2} = 1b_j
\] (23)
The matrices have following form:

\[ A_{j-2,j}^T = -A_{j-1,j}^{-1} A_{j-1,j-1} A_{j-2,j-1} \]
\[ A_{j,j} = A_{j,j}^{-1} - A_{j-1,j}^{-1} A_{j-1,j-1} A_{j-1,j}^{-1} A_{j,j+1}^{-1} A_{j,j+1}^T \]
\[ A_{j,j+2} = -A_{j,j+1}^{-1} A_{j,j+1}^{-1} A_{j,j+2} \]
\[ b_j = b_j - A_{j-1,j}^{-1} b_{j-1} - A_{j,j+1}^{-1} b_{j+1} \]

The left-standing superscript denotes the level of the elimination. The reduction of the size can be repeated leading to the systematic downsizing of the system by factor 2. In the situation \( j=1 \) (the first subsystem), the expression \( A_{1,j-2}^T \mu_{j-2} \) vanishes from (23) and similar for \( j=n \) (the last subsystem) the expression \( A_{1,j+2} \mu_{j+2} \) vanishes. In the end, the root system is obtained with one element only, which is solved. The whole procedure of the variables elimination is depicted in Fig. 5 for the example of system with 16 sub-blocks, where the grey colour represents the variables being eliminated on the individual elimination level.

After that process the procedure of back-substitution has to be carried out in order to evaluate the expressions (22) and thus to solve the whole system. It is important, that the substitution phase and the back-substitution can be carried out in parallel, evaluating the substitutions in the same time. Thus a procedure is obtained, in which one level of the elimination has a constant time. The size of the system being solved is reduced to a half by one elimination level. Therefore the number of the levels is:

\[ N_L = \log_2(n) \]

It is worth to stress out, that the substitution phase represents math operations as the inversion of small matrices \( M_{ii} \) (naturally 3x3 dimension) and the small-dimensional matrices and vectors multiplications and adding. The whole procedure represents thus just the matrix transformations.

3 THE COMBINATION WITH THE CHOLESKY DECOMPOSITION

As stated before, the structure of the key linear system, which has to be solved, is sparse, band, symmetric positive definite. Therefore the system is extremely suitable for Cholesky decomposition. The experimental results have showed, that this procedure is even more effective for the solution of smaller systems than the parallel elimination described above. However, there is a limit in the number of bodies (ca. 1000 ), for which the Cholesky decomposition ceases to be the leader, see Fig. (6). The effective implementation of the Cholesky decomposition called Cholmod [17] has been used for the testing in the Matlab environment.

It is obvious from the comparison of both method, that their combination can be more effective, then the usage of single method only. Therefore, following case has been tested for kinematic chains with different numbers of bodies: The elimination procedure has been used to reduce the system size to a certain limit \( L \) denoting the number of elements in the reduced system.
Figure 6. The comparison of the complexity of the solution by the elimination procedure ($\log_2 N$) and by the Cholesky decomposition (Cholmod).

Then this reduced system has been solved using the Cholmod procedure. The results compared with the original implementation of the elimination are depicted in fig. 7.

Figure 7. The comparison of the complexity of the solution by the elimination procedure ($\log_2 N$) and by the combination of the elimination with Cholesky decomposition ($\log_2 N \text{ modif}$).

It is obvious from the results, that it is more effective for the larger systems to use the elimination procedure to reduce the system corresponding to approximately 200 bodies and then to use the Cholmod procedure.

The abovementioned tests have been emulated for the case of the natural block decomposition corresponding to the structure of the system. Thus the small blocks (3x3) are considered. Moreover, the assumption has been considered, that the number of computing processors is the same as the number of the particular bodies in the system. This is, however, an ideal case. The number of bodies being modelled is usually greater than the number of processors being at disposal.

Therefore this situation have been investigated. In the case, that the 3x3 division is used and there are not enough processors for the matrix transformations, the process is following. The number of subsystems $n_s$ for elimination is the same as the number of bodies $n$. The number of processors $n_p$ is smaller than $n$. Therefore it is possible to evaluate only $n_p$ elimination in parallel on one elimination level and the rest of them has to be carried out after that. The results have showed, however, that it is more convenient to do the division into the blocks in the way, that the number of subsystems is the same as the number of processors, $n_s = n_p$. The size of the
subsystem is greater, however, the Cholesky decomposition is applied again for their effective solution. Thus the optimal elimination process is obtained, see Fig. 8. The complexity of the solution is depicted for the non-optimal 3x3 division (log₂ N) and for the optimal one (log₂ N \text{ optimal}) for different factors $K = \frac{n}{n_p}$ representing the relation between number of bodies in the multibody system and the number of processors. It is obvious, that the red dash-curves representing the optimal division lie near the optimal case $K = 1$ (the same number of bodies and processors).

![Figure 8. The comparison of the optimal and the non-optimal division](image)

4 CONCLUSIONS

The elimination method described in this paper represents a procedure for the multibody dynamics solution with logarithmic complexity. The natural coordinates are used for the description of the kinematic chain. And as the result of it, the EOMs are assembled, which structure is sparse, band, symmetric, positive definite. The Modified State Space method have been adopted for the EOM assembling. The elimination method is qualitatively similar to the DAC procedures. However, the method can be combined with effective implementation of the Cholesky decomposition and thus to be more effective than the simple application. Thus this elimination method yields new advantages. One of them is the fact that the substitution process, which is the analogy to the assembly process in DAC, has not to be carried out until the root level is achieved. It is more convenient to carry out the elimination to a limit system size, for which the Cholesky decomposition is further applied. Another advantage of this elimination method is the scalability. It is not necessary to divide the whole system of equations into the subsystems corresponding to the particular bodies, as by DAC. The division can be carried out in an optimal way, in which the number of subsystems corresponds to the number of processors available for the computing. Thus the greater effectivity is obtained for the solution of the whole system of equations. These features can make this method favourable in comparison with the standard DAC procedures. The numerical experiments have been carried in the Matlab environment emulating the parallel implementation, any barriers are, however, not known, which disable the real parallel implementation. The application for the simple kinematic chain have been presented in this paper. Nowadays the extension for the branched multibody systems with loops is being prepared.
REFERENCES


Various multibody dynamic models for the description of plane Kirchhoff rods

Holger Lang#, Hannah Laube#, Sigrid Leyendecker#

# Chair of Applied Dynamics
University of Erlangen-Nuremberg
Haberstrasse 1, 91058 Erlangen, Germany
[holger.lang, hannah.laube, sigrid.leyendecker]@fau.de

ABSTRACT
Three modeling concepts from multibody dynamics (absolute redundant coordinates, minimal relative coordinates and mixed coordinates) are applied to a geometrically exact discrete viscoelastic Kirchhoff rod model from continuum mechanics. We compare the numerical effort and accuracy, using BDF time integration schemes, established in multibody dynamics.

Keywords: Multibody formalism, flexible multibody dynamics, BDF time integration, geometrically exact viscoelastic Kirchhoff rods, absolute & relative coordinates.

1 INTRODUCTION
The plane Kirchhoff rod model is well known in continuum mechanics for the dynamic simulation of slender structures. It is a geometrically exact generalisation of the linear Euler-Bernoulli beam that takes into account extensional and bending deformations [4]. In contrast to a Reissner rod (i.e. a plane Cosserat rod [9]), the plane and rigid cross sections always stay perpendicular to the centerline of mass centroids. Therefore, it does not incorporate transverse shear strains and is well suited for very slim structures. Viscoelasticity is taken into account by the use of Kelvin-Voigt material.

We consider a Finite Element discretisation that is based on the discrete Kirchhoff beam kinematics displayed in Figure 1. It is a two-dimensional plane restriction of the one proposed in [4, 5] for three-dimensional Kirchhoff and Cosserat rods on a staggered grid.

The proper choice of coordinates plays a crucial role concerning accuracy and numerical complexity during time integration. Here, we compare three approaches, which are well known in multibody dynamics simulations, and apply them to the proposed FE model. In what follows, \( N \) denotes the number of rod elements.

- The first approach is to use redundant absolute coordinates \( p \in \mathbb{R}^{4N} \) and to enforce shear rigidity via holonomic constraints. Index reduction is applied.

- The second is to use relative (and minimal) joint coordinates \( q \in \mathbb{R}^{2N} \). In the context of continuum mechanics, these magnitudes correspond to the extensional strains and the bending curvatures. This means that strain-like variables are used as the primary unknowns instead of the positions or displacements.

- The third approach is to use \( q \in \mathbb{R}^{2N} \) as the primary unknowns, but – at the same time – use both \( q \) and \( p \) within the dynamic right hand side function in order to solve the dynamic balance in every time step with linear complexity. This formulation constitutes a simple form of an \( \mathcal{O}(N) \)-multibody formalism.

The second and third version are not standard in continuum mechanics. The main intention to use these is that the time stepping characteristic should be significantly improved in comparison to the first approach. We compare the three approaches concerning accuracy and numerical effort,
focussing on the numerical linear algebra involved. Time integration is performed with a standard BDF method in MATLAB [10].

**Nomenclature** Throughout the paper, for a complex number \( z = u + iv \), where \( u, v \in \mathbb{R} \) and \( i \) denotes the imaginary unit, satisfying \( i^2 = -1 \), we identify

\[
\begin{align*}
z &= u + iv \simeq \begin{bmatrix} u \\ v \end{bmatrix}, \\
\bar{z} &= u - iv \simeq \begin{bmatrix} u \\ -v \end{bmatrix} \quad \text{and} \quad iz = iu - v \simeq \begin{bmatrix} -v \\ u \end{bmatrix}
\end{align*}
\]

with their vector counterparts in \( \mathbb{R}^2 \). Therefore, they are written in bold letters. Further, for two vectors \( \mathbf{x}, \mathbf{y} \in \mathbb{R}^2 \), \( \langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^\top \mathbf{y} \) denotes the Euclidean scalar product of \( \mathbf{x} \) and \( \mathbf{y} \). Likewise, \( \| \mathbf{x} \| = \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle} = \sqrt{\mathbf{x}^\top \mathbf{x}} \) is the Euclidean norm of \( \mathbf{x} \in \mathbb{R}^2 \). For a regular matrix \( A \in \mathbb{R}^{n \times n} \) and a vector \( \mathbf{x} \in \mathbb{R}^n \) we let \( A \backslash \mathbf{x} = A^{-1} \mathbf{x} \) denote the left-division of \( \mathbf{x} \) by \( A \) in order to emphasise the fact that the inverse \( A^{-1} \) is never numerically computed. In fact, we always use Gaussian \( LU \) decomposition \( A = LU \) in combination with forward and backward substitution, i.e. \( A \backslash \mathbf{x} = U \backslash (L \backslash \mathbf{x}) \).

## 2 CONTINUUM MODEL

The kinematics of a (plane) Reissner rod is completely determined by its centerline \( \mathbf{x} : [0, L] \times \mathbb{R} \to \mathbb{R}^2 \), \( (s,t) \mapsto \mathbf{x}(s,t) \) and its unit complex number field \( \mathbf{z} : [0, L] \times \mathbb{R} \to \mathbb{S}^1 \), \((s,t) \mapsto \mathbf{z}(s,t)\), indicating the cross section unit normal. Here \( \mathbb{S}^1 = \{ z \in \mathbb{C} : \|z\|^2 = 1 \} \) denotes the complex unit circle, \( s \) denotes the arc-length parameter in the straight, undeformed configuration, \( t \) is the time. The basic deformation modes of a Reissner rod are extension (i.e. axial dilation), bending and transverse shearing. See [9].

For a plane Kirchhoff rod, there is the additional Euler-Bernoulli restriction \( \langle iz, \partial \mathbf{x}/\partial s \rangle = 0 \), which means that the cross section directors \( iz \) are always orthogonal to the centerline tangents \( \partial \mathbf{x}/\partial s \) during deformation [4]. Equivalently, the cross section normals \( \mathbf{z} \) equal the unit tangents \( \| \partial \mathbf{x}/\partial s \|/\| \partial \mathbf{z}/\partial s \| \) of the centerline. The cross sections are assumed to be plane and rigid. The basic deformation modes of a plane Kirchhoff rod are extension and bending.

![Figure 1](image.png)

**Figure 1.** Plane Kirchhoff beam kinematics. The directors \( \mathbf{z}_n \) are parallel to the discrete centerline tangents \( \mathbf{x}_n - \mathbf{x}_{n-1} \). The directors \( iz_n \) indicate the orientation of the cross section, perpendicular to the centerline tangents. The position \( \mathbf{x}_0 \) and orientation \( \mathbf{z}_0 \) at the left are prescribed as boundary values.

The **internal elastic energy** for a plane Kirchhoff rod is given by

\[
\mathcal{V} = \frac{1}{2} \int_{0}^{L} EA \left( \| \frac{\partial \mathbf{x}}{\partial s} \| - 1 \right)^2 ds + \frac{1}{2} \int_{0}^{L} EI \mathcal{S} \left( \frac{\partial \mathbf{z}}{\partial s} \right)^2 ds, \tag{1}
\]

see [4, 5]. Here, \( (EA)(s) \) denotes the extensional (or axial) stiffness with Young’s modulus \( E(s) \) and the cross section area \( A(s) \), \( (EI)(s) \) is the bending stiffness with the geometric moment of inertia \( I(s) \). \( \| \partial \mathbf{x}/\partial s \| - 1 \) is the extensional strain, \( K = \mathcal{S} (\partial \mathbf{z}/\partial s) \) is the bending curvature.
It is assumed that the viscous power is proportional to the strain and curvature rates, which corresponds to the internal dissipative potential

$$\mathcal{D} = \frac{1}{2} \int_0^L \eta_E A \left( \frac{\partial}{\partial t} \| \frac{\partial \mathbf{x}}{\partial s} \| \right)^2 \, ds + \frac{1}{2} \int_0^L \eta_I \left( \frac{\partial}{\partial t} \mathbf{3} \left( \frac{\partial \mathbf{z}}{\partial s} \right) \right)^2 \, ds. \quad (2)$$

It was shown in [6] that (1) in combination with (2) corresponds to Kelvin-Voigt viscoelastic material with the extensional viscosity $\eta_E(s)$.

The internal kinetic energy is given by

$$\mathcal{F} = \frac{1}{2} \int_0^L \rho A \left\| \frac{\partial \mathbf{x}}{\partial t} \right\|^2 \, ds + \frac{1}{2} \int_0^L \rho I \mathbf{3} \left( \frac{\partial \mathbf{z}}{\partial t} \right)^2 \, ds \quad (3)$$

with the mass density $\rho(s)$. Note that $\| \partial \mathbf{x} / \partial t \|$ is the centroid velocity, $\Omega = \mathbb{I} \partial \mathbf{z} / \partial t$ the scalar angular velocity of the cross section. The derivation of the complex expressions for $K$ and $\Omega$ can be carried out in an analogous way as for their quaternionic counterpart, see [4, 5].

**Remark** Let $SO(2) = \{ \mathbf{R} \in \mathbb{R}^{2 \times 2} : \mathbf{R}^T \mathbf{R} = \mathbf{I}, \text{det} \mathbf{R} = 1 \}$ denote the manifold of plane rotations. Then, due to the diffeomorphic, structure-preserving Euler-Gauss-mapping $\mathbb{S}^1 \ni \mathbf{z} \mapsto \mathbf{R}(\mathbf{z}) = [\mathbf{z} \, | \bar{\mathbf{z}}] \in SO(2)$, the unit circle $\mathbb{S}^1$, equipped with its complex structure, is isomorphic to $SO(2)$, considered both as manifolds and groups. We use complex numbers to enable the extension to a quaternionic formulation for Kirchhoff beams in 3D space later. □

### 3 DISCRETISATION IN ABSOLUTE COORDINATES

We propose the following discretisation scheme on a staggered grid [5]. The arclength parameter interval $[0, L]$ is discretised into $N$ segments with nodes

$$0 = s_0 < s_1 < \ldots < s_{N-1} < s_N = L.$$ 

The segment midpoints are denoted by $\sigma_n = \frac{1}{2}(s_{n-1} + s_n)$. We let $\Delta s_n = s_n - s_{n-1}$ denote the length of the straight segment $[s_{n-1}, s_n]$ and $\delta s_0 = \frac{1}{2}\Delta s_1$, $\delta s_N = \frac{1}{2}\Delta s_N$ denote the length of the buckled segment $[s_{N-1}, s_N]$. We restrict ourselves to an autonomous system with fully clamped boundary on the left and a free boundary on the right. This makes the notational exposition as simple as possible. See Figure 1 for the discrete kinematics.

The first approach uses redundant absolute coordinates $\mathbf{p} = (\mathbf{z}_1, \mathbf{x}_1, \ldots, \mathbf{z}_N, \mathbf{x}_N) \in \mathbb{R}^{4N}$, where the $\mathbf{x}_n(t) \approx \mathbf{R}(\mathbf{z}_n,t) \in \mathbb{R}^2$ are approximations of the absolute translations (i.e. the cross section centroids) on the nodes and $\mathbf{z}_n \approx \mathbf{z}(\sigma_n,t) \in \mathbb{S}^1$ are approximations of the absolute rotations (i.e. the cross section orientations) on the segment midpoints.

We introduce the discrete curvature $\delta_n$ (belonging to the $n$-th node) resp. the discrete extensional strain $\xi_n$ (belonging to the $n$-th segment midpoint) by

$$\delta_n = \mathbb{I}(\mathbf{z}_{n-1} - \mathbf{z}_n) \quad \text{resp.} \quad \xi_n = \| \mathbf{x}_n - \mathbf{x}_{n-1} \| \quad \text{for} \quad n = 1, \ldots, N. \quad (4)$$

We let $\mathbf{z} = (\mathbf{z}_1, \ldots, \mathbf{z}_N) \in \mathbb{R}^N$ and $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_N) \in \mathbb{R}^N$. The internal elastic energy $\mathcal{V}$ in (1) is discretised via the midpoint and trapezoidal rule,

$$\mathcal{V} \approx V(\mathbf{p}) = \frac{1}{2} \sum_{n=1}^N \frac{E A}{\Delta s_n} \left( \| \mathbf{x}_n - \mathbf{x}_{n-1} \| - \Delta s_n \right)^2 + \frac{1}{2} \sum_{n=0}^N \frac{E I}{\Delta s_n} \mathbb{I}(\mathbf{z}_{n-1} - \mathbf{z}_n)^2$$

$$= \frac{1}{2} \sum_{n=1}^N \frac{E A}{\Delta s_n} \left( \xi_n(\mathbf{x}) - \delta s_n \right)^2 + \frac{1}{2} \sum_{n=0}^N \frac{E I}{\Delta s_n} \delta_n(\mathbf{z})^2. \quad (5)$$

(Not that – as the right end at $s = L$ is supposed to be free – the bending curvature and the internal bending moment vanish identically at $s = L$.) Likewise for the internal dissipative potential $\mathcal{D}$ in (2),

$$\mathcal{D} \approx \mathcal{D}(\mathbf{p}, \dot{\mathbf{p}}) = \frac{1}{2} \sum_{n=1}^N \left( \frac{\eta E A}{\Delta s_n} \right) \xi_n(\mathbf{x}, \dot{\mathbf{x}})^2 + \frac{1}{2} \sum_{n=0}^N \left( \frac{\eta I}{\Delta s_n} \right) \delta_n(\mathbf{z}, \dot{\mathbf{z}})^2. \quad (6)$$

154
For the internal kinetic energy $\mathcal{T}$ in (3), we let

$$\mathcal{T} \approx T(p, \dot{p}) = \frac{1}{2} \sum_{n=0}^{N} \left( \rho \frac{\partial A}{\partial s} \right)_n \| \dot{x}_n \|^2 + \frac{1}{2} \sum_{n=1}^{N} \left( \rho \left( \frac{\partial G}{\partial A} \right)_n \right) \dot{z}_n \dot{z}_n^T. \quad (7)$$

The unity condition for $z_n$ and the shear rigidity, i.e. the orthogonality of the cross section w.r.t. the centerline, lead to internal holonomic constraints of the form $0 = g(p) = (g_1, \ldots, g_N) \in \mathbb{R}^{2N}$, where

$$g_n(x_{n-1}, z_n, x_n) = \begin{bmatrix} ||z_n||^2 - 1 \\ \langle i z_n, x_n - x_{n-1} \rangle \end{bmatrix} \quad \text{for} \ n = 1, \ldots, N. \quad (8)$$

Applying the Lagrange-d’Alembert principle, we arrive at the constrained Euler-Lagrange equations of motion

$$\frac{d}{dt} \left( \frac{\partial T}{\partial \dot{p}} \right) + \frac{\partial V}{\partial p} + \frac{\partial D}{\partial p} + G^T \lambda = 0, \quad g = 0$$

with the d’Alembert constraint forces $-G(p)^T \lambda$, where $G(p) = \nabla g(p) \in \mathbb{R}^{2N \times 4N}$ is the Jacobian of $g(p)$ and $\lambda \in \mathbb{R}^{2N}$ are Lagrange multipliers, which are measuring the magnitude of the constraint forces.

The first constraint force is somewhat artificial due to the parametrisation of rotations by complex numbers. It is necessary to keep $z_n$ on its spherical orbit. It is not hard to see that the second one represents the transverse shear force. (Note that here – in contrast to the shear flexible Reissner rod – the transverse shear forces are not of constitutive, but of reactive kind.) Then, the standard index 1 version of the equations of motion takes the form

$$\begin{bmatrix} \dot{p} \\ \dot{v}_p \\ \dot{\lambda} \end{bmatrix} = \begin{bmatrix} M(p) & G(p)^T \\ G(p) & 0 \end{bmatrix} \begin{bmatrix} f(p, v_p) \\ -G(p, v_p) v_p \end{bmatrix}, \quad (9)$$

with the state dependent mass $M(v_p) = \partial^2 T(p, v_p)/\partial v_p^2 \in \mathbb{R}^{4N \times 4N}$, being the Hessian of the quadratic kinetic energy, and the generalised force function

$$f(p, v_p) = M(p, v_p) v_p - \frac{\partial V}{\partial p}(p) - \frac{\partial D}{\partial v_p}(p, v_p), \quad (10)$$

which is of dimension $4N$, incorporating the internal forces and momenta. Due to the parametrisation of rotations by complex numbers, $f$ is free of trigonometric expressions and therefore fast to evaluate. (Further external actuating forces or momenta might simply be added.) System (9), composed with the projection $(\dot{p}, \dot{v}_p, \dot{\lambda}) \mapsto (\dot{p}, \dot{v}_p)$ yields a first order ODE system

$$\dot{u} = F_A(u) \quad \text{for the unknowns} \quad u = \begin{bmatrix} p \\ v_p \end{bmatrix} \quad (11)$$

of dimension $8N$. The index ‘A’ stands for ‘absolute coordinates’. The structure of the $6N \times 6N$ system matrix in (9) is block-banded, see Figure 2. After rearranging the unknowns according to the chain topology of the discrete beam model, it becomes banded with a bandwidth $5$, see Figure 2. We recall the fact from numerical linear algebra, that the numerical effort – measured in the total number of multiplikations and divisions – to solve a linear system of dimension $n$ with bandwidth $b$ is of order $O(b^2n)$ for large $n$, see [3]. Therefore, solving the linear system of equations in (9) in each time step grows with complexity $5^2 \cdot 6N = 150N$.

**Remark** The proposed discretisation scheme is of second order in space, if the discretisation is chosen equidistant [4]. For a thorough discussion, why finite quotients for $w_n$ in (4) constitute a reasonable choice to discretise curvature, we refer to [5] for the quaternionic case in 3D. 

\[\square\]
4 STRAIN COORDINATES

The second approach uses ‘strain coordinates’ \( q = (w_1, \xi_1, \ldots, w_N, \xi_N) \in \mathbb{R}^{2N} \), where the discrete bending curvatures \( w_n \) resp. discrete extensional strains \( \xi_n \) from (4) represent the relative rotations resp. relative translations. In the nomenclature of multibody dynamics, they are usually called minimal ‘joint’ (or ‘relative’) coordinates.

We write \( q = \psi(p) \) to denote the forward transformation (4), mapping the absolute onto the relative strain coordinates. (This transformation is not necessarily of recursive kind.) Its inverse, the backward transformation \( p = \varphi(q) \), is recursively given by

\[
\begin{align*}
\mathbf{z}_n &= \left( \sqrt{1 - w_n^2 + iw_n} \right) \mathbf{z}_{n-1} \quad \text{resp.} \quad \mathbf{x}_n = \mathbf{x}_{n-1} + \xi_n \mathbf{z}_n \quad \text{for} \quad n = 1, \ldots, N.
\end{align*}
\]

(12)

That way, the holonomic constraints (8) are clearly satisfied. System (9) is then transformed to the analytically equivalent form

\[
\begin{align*}
\mathbf{q} &= \mathbf{v}_q \\
\mathbf{M}(q) \dot{\mathbf{v}}_q &= \mathbf{\Phi}(q)^\top \{ \mathbf{f}(p, \mathbf{v}_p) - \mathbf{M}(p) \mathbf{\Phi}(q) \mathbf{v}_q \} \bigg|_{p = \varphi(q), v_p = \Phi(q) v_q}
\end{align*}
\]

(13)

with the Jacobian \( \mathbf{\Phi}(q) = \nabla \varphi(q) \), the velocity transformation \( \mathbf{p} = \mathbf{\Phi}(q) \mathbf{q} \) and the \( 2N \times 2N \) minimal mass matrix

\[
\mathbf{M}(q) = \mathbf{\Phi}(q)^\top \mathbf{M}(p) \mathbf{\Phi}(q) \bigg|_{p = \varphi(q)}.
\]

(14)

It is depicted in Figure 3. The constraint on velocity level \( \mathbf{0} = \mathbf{G}(p) \dot{\mathbf{p}} = \mathbf{G}(\varphi(q)) \mathbf{\Phi}(q) \dot{\mathbf{q}} \) and the independence of the coordinates \( \mathbf{q} \) imply that the columns of \( \mathbf{\Phi}(q) \) span the null space (or kernel) of \( \mathbf{G}(\varphi(q)) \). Especially, \( \mathbf{G}(\varphi(q)) \mathbf{\Phi}(q) \equiv \mathbf{0} \), which is used to eliminate the constraint forces \( \mathbf{G}(p)^\top \lambda \). Due to the recursive structure of (12), the Jacobian \( \mathbf{\Phi}(q) \) has a triangular shape as displayed in Figure 3.

System (13) with the minimal mass (14) yields the first order ODE system

\[
\mathbf{A}_S(u) \dot{\mathbf{u}} = \mathbf{F}_S(u), \quad \text{where} \quad \mathbf{A}(u) = \begin{bmatrix} \mathbf{E} & \mathbf{0} \\ \mathbf{0} & \mathbf{M}(q) \end{bmatrix} \quad \text{for the unknowns} \quad \mathbf{u} = \begin{bmatrix} \mathbf{q} \\ \mathbf{v}_q \end{bmatrix}
\]

(15)
of dimension $4N$. We use the index ‘$S$’ for ‘strain coordinates’. By the use of BDF multistep methods, it is not necessary to solve (13) for $\dot{v}_q$ in each time step. However, the effort in linear algebra grows like $O(N^2)$, since the minimal mass $M(q)$ is fully populated. See Figure 3, Table 2 and [1, 3].

**Remark** Alternatively to the expression in (4), the extensional strains $\dot{e}_n$ might be written in the form

$$\dot{e}_n = \|x_n - x_{n-1}\| = \|z_n\|\|x_n - x_{n-1}\| \cos \angle (z_n, x_n - x_{n-1}) = (z_n, x_n - x_{n-1}),$$

(16)

since $\|z_n\| = 1$ and $\angle (z_n, x_n - x_{n-1}) = 0$ because of the constraints (8). Note that the director $z_n$ is parallel to the discrete centerline tangent $x_n - x_{n-1}$. Using (16) instead of (4), the dynamic right-hand-side generalised force function $f$ in (10) is free of any algebraic (e.g. square root) and any trigonometric function.

## 5 MIXED COORDINATES

The third approach is formulated in terms of *mixed coordinates*, as it is called in [1]. Adding the equation $0\ddot{q} = 0 \in \mathbb{R}^{2N}$ and augmenting system (9) by the dummy Lagrange multiplier $\eta = 0 \in \mathbb{R}^{2N}$, the index 1 system

$$\begin{cases}
\dot{p} = v_p \\
\dot{q} = v_q \\
\dot{v}_p = \begin{bmatrix}
M(p) & 0 & G(p)^T & -\Psi(p)^T \\
0 & 0 & 0 & E \\
G(p) & 0 & 0 & 0 \\
-\Psi'(p) & E & 0 & 0
\end{bmatrix}
\begin{bmatrix}
f(p, v_p) \\
0 \\
-G(p, v_p)v_p \\
\Psi'(p, v_p)v_p
\end{bmatrix}
\end{cases}
$$

(17)

with the Jacobian $\Psi'(p) = \nabla \Psi(p)$ is obtained. It is analytically equivalent to (9). System (17), composed with the projection mapping $(p, q, v_p, v_q, \lambda, \eta) \mapsto (\dot{q}, \dot{v}_q)$ yields the first order ODE system

$$\dot{u} = F_M(u) \quad \text{for the unknowns} \quad u = \begin{bmatrix} q \\ v_q \end{bmatrix}
$$

(18)

of minimum dimension $4N$. The index ‘$M$’ stands for ‘mixed coordinates’. Clearly, from the analytical viewpoint, $F_S \equiv F_M$.

The idea behind the ‘mixed’ coordinate formulation is the sparsity of the system matrix in (17). Solving the linear system of equations in (17) in each time step grows with complexity $16^2 \cdot 10N = 2560N$, cf. [3], as the structure is block-banded or banded with bandwidth 16, see Figure 4.
bandwidth is much larger than the one in (9), but we point out that Gaussian elimination takes place not outside the so-called ‘hull’ [8]. Therefore, the estimate $2560N$ is rather pessimistic. Alternatively, the linear system in (17) might be solved iteratively as a sequence of $N$ ‘small’ linear systems [1].

**Remark** Solving the index 1 system (9) numerically, there is a linear drift-off in the velocity variables $v_p$, yielding a quadratic drift of the position variables $p$ from the configuration manifold. As the coordinates $q$ are minimal, there is no drift-off phenomenon, when using formulation (13) or (17).

6 RESULTS, NUMERICAL STATISTICS AND TASK

The discussion of pros and cons for each of the formulations (9, 11), (13, 15) and (17, 18) with respect to numerical effort (e.g. function calls and time stepsizes) and accuracy is part of this section.

As an example, we consider the dynamic scenario depicted in Figure 5. It is a flexible Kirchhoff pendulum rod, made of rubber-like material, fully clamped at the left end and dynamically swinging under its own gravity load. The parameters used can be found in Table 1.

<table>
<thead>
<tr>
<th>$L$</th>
<th>1.00m</th>
<th>$r$</th>
<th>$1.00 \cdot 10^{-2}m$</th>
<th>$A$</th>
<th>$3.14 \cdot 10^{-4}m^2$</th>
<th>$I$</th>
<th>$7.85 \cdot 10^{-9}m^4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$E$</td>
<td>$5.00 \cdot 10^9$ kg m$^{-1}$s$^{-2}$</td>
<td>$\eta_E$</td>
<td>$8.44 \cdot 10^9$ kg m$^{-1}$s$^{-1}$</td>
<td>$\rho$</td>
<td>$1.10 \cdot 10^3$ kg m$^{-3}$</td>
<td>$g$</td>
<td>$9.81$ ms$^{-2}$</td>
</tr>
</tbody>
</table>

**Table 1.** Material and geometry parameters, where $A = \pi r^2$, $I = \frac{2}{3} r^4$ for a circular cross section of radius $r$. ($\eta_E$ corresponds to 10% of the critical extensional viscosity.)

Figure 7 displays the essential statistical results that are typical in conjunction with the MATLAB time integrators ODE15s, a BDF multistep method of variable order, reaching from 1 to 5. For
Figure 5. Snapshot movie of a swinging plane Kirchoff rod, made of rubber-like material. It is fully clamped at the left end and free at the right end. Plotted are the centerline centroids \( x_n = (x_n, y_n) \) and the cross section directors \( i_z_n \), which are perpendicular to the discrete tangents \( x_n - x_{n-1} \), for \( N = 10 \).

Equidistant time stepping with stepsize \( \Delta t \), they read

\[
\begin{align*}
\text{BDF 1:} & \quad A(u_i) \left( u_i - u_{i-1} \right) = \Delta t F(u_i) \\
\text{BDF 2:} & \quad A(u_i) \left( \frac{3}{2} u_i - 2u_{i-1} + \frac{1}{2} u_{i-2} \right) = \Delta t F(u_i) \\
\text{BDF 3:} & \quad A(u_i) \left( \frac{11}{6} u_i - 3u_{i-1} + \frac{3}{2} u_{i-2} - \frac{1}{3} u_{i-3} \right) = \Delta t F(u_i) \\
\text{BDF 4:} & \quad A(u_i) \left( \frac{25}{12} u_i - 4u_{i-1} + 3u_{i-2} - \frac{4}{3} u_{i-3} + \frac{1}{4} u_{i-4} \right) = \Delta t F(u_i) \\
\text{BDF 5:} & \quad A(u_i) \left( \frac{137}{60} u_i - 5u_{i-1} + 5u_{i-2} - \frac{10}{3} u_{i-3} + \frac{5}{4} u_{i-4} - \frac{1}{5} u_{i-5} \right) = \Delta t F(u_i)
\end{align*}
\]

BDFs are of widespread use in multibody dynamics [1], because they are well suited for stiff systems and for systems of the form

\[
A(u) \dot{u} = F(u)
\]

with a regular – or even singular – state dependent, square matrix \( A(u) \). Each of the systems (11) with \( A = E \), (15) with \( A = A_S \) and (18) with \( A = E \) is of that form, \( A(u) \) being regular for each \( u \). In each time step one of the schemes in (19) is solved for the new state \( u_i \) at time \( t_i \) with a simplified Newton-Raphson method, which corrects the predictor. Here, the iteration matrix

\[
J(u_i) = aA(u_i) + \nabla \{ A(\ast)(a u_i + \ldots) \} \big|_{\ast = u_i} = \Delta t \nabla F(u_i)
\]

is used, the parameter \( a \in \{ 1, 3/2, 11/6, 25/12, 137/60 \} \) denoting the leading coefficient according to the five BDF schemes in (19). In the experiments presented below, both Jacobians needed in (20) are approximated by finite differences.

In almost each statistical criterion, the mixed formulation turns out to be superior compared to both the redundant and the strain coordinate formulation, cf. Figure 7. This is reflected as well in
Figure 6. Time stepsizes. By the use of minimal coordinates \( q \) in ‘\( S \)’ and ‘\( M \)’, the stepping characteristics reflect the dynamics of the scenario in Figure 5 perfectly due to the absence of redundance.

the relative computational times. The NDF schemes [10], which are as well supplied in ODE15s, perform slightly better. But the performance gain is almost not worth mentioning.

Since the computational times in MATLAB are not really objective (compared to codes from compiler languages as C, C++ or Fortran77), we estimate the total numerical effort, focussing on the numerical linear algebra performed within the right-hand side functions \( F(u) \), the left-hand-side functions \( A(u)u \) and during the corrector iterations within the core integrator. The effort to compute \( f(p, \dot{p}) \), \( G(p) \), \( G(p, \dot{p}) \), \( M(p) \), the forward and backward recursions via \( \psi \) and \( \phi \) within the right-hand sides – each linearly increasing – is extremely low and is neglected.

<table>
<thead>
<tr>
<th>Formulation</th>
<th>absolute ( \cdot \leftarrow A )</th>
<th>strain ( \cdot \leftarrow S )</th>
<th>mixed ( \cdot \leftarrow M )</th>
</tr>
</thead>
<tbody>
<tr>
<td># r. h. s. function ( F(u) ) evaluations</td>
<td>( 5^2 \cdot 6N )</td>
<td>( 8N^2 )</td>
<td>( 16^2 \cdot 10N )</td>
</tr>
<tr>
<td># r. h. s. Jacobians ( \partial F(u)/\partial u ) evaluations</td>
<td>( (8N)^2 )</td>
<td>( (4N)^2 )</td>
<td>( (4N)^2 )</td>
</tr>
<tr>
<td># l. h. s. function ( A(u)u ) evaluations</td>
<td>0</td>
<td>12N(^2)</td>
<td>0</td>
</tr>
<tr>
<td># LU decompositions of ( J(u) )</td>
<td>( (8N)^3 )</td>
<td>( (4N)^3 )</td>
<td>( (4N)^3 )</td>
</tr>
<tr>
<td># forward &amp; backward substitutions</td>
<td>( (8N)^2 )</td>
<td>( (4N)^2 )</td>
<td>( (4N)^2 )</td>
</tr>
</tbody>
</table>

Table 2. Numerical effort in terms of r. h. s. function calls, r. h. s. Jacobian calls, l. h. s. function calls and numerical linear algebra.

We measure the numerical linear algebra ‘task’ (or ‘effort’) classically in terms of the total amount of essential operations, i.e. multiplications and divisions. Table 2 summarises, which of the five essential statistical indicators (number of r. h. s., l. h. s. function and Jacobian evaluations, number
of LU decompositions of $J$ and number of forward & backward substitutions), is punished with which amount of essential operations. Four of those five indicators are depicted in Figure 7 as a function of $N$. The total numerical linear algebra task is the total sum of these essential operations.

Figure 8, depicting the total task, clearly demonstrates the superiority of choosing minimal coordinates (either ‘S’ or ‘M’). This is certainly due to the fact that the solver is not forced to decrease the time step sizes because of mechanical redundance, see Figure 6. Clearly, this is the main reason to use them. For a large number $N$ of elements, ‘M’ performs better than ‘S’. In addition to the increase in statistical indicators in Figure 7, the reason is that the r.h.s. function in (17) can be evaluated with linear complexity instead of (13), where the growth is quadratic.

A drawback of all three formulations proposed is the fact that the effort for numerical linear algebra during the simplified Newton corrector iterations within the core solver explodes for large $N$. For applications with coarse accuracy requirements, i.e. small $N$, such as simulation of cables and hoses in industrial assemblies, the proposed formulations in connection with BDF methods definitively have realtime capability. Possible remedies for large $N$ are the following.

- The Jacobian $J$ in (20) has block-banded structure. By the use of especially adapted block-banded Gaussian LU-decomposition the numerical effort might be reduced, provided that the blocks can be decomposed iteratively in a ‘save’ fashion [3].

- Further, the Jacobians for ‘A’ and ‘M’ display a certain kind of diagonal dominance. Therefore, a simplified quasi-Newton method, cutting off the diagonals with ‘small’ entries, seems to be auspicious.

If a large number of elements is required, however, one certainly cannot help but solving (9) as an index 1 DAE, not discarding the Lagrange multipliers.

For comparison, in Figure 8, we additionally included the total numerical task for ODE45, an explicit Runge-Kutta method based on the embedded scheme of Dormand and Prince [10]. Clearly, although there are no shearing oscillations/degrees of freedom contained, the Kirchhoff model is highly numerically stiff, as it emerges from continuum mechanics. Therefore – as it is for the shear flexible Reissner model [4] –, each explicit method suffers from stringent stepsize restrictions in order to run stable.

7 CONCLUSIONS

We present a discretisation of a two-dimensional plane discrete Kirchhoff rod model. The proposed choice of ‘strain’ or ‘mixed’ minimal coordinates yield promising results in accuracy and numerical task, which is worth to be examined further. The extension to discrete quaterionic Kirchhoff rods in three dimensions is straightforward and part of forthcoming work.

REFERENCES

Figure 7. Statistics in conjunction with ODE15S in MATLAB: Number of time steps, function and Jacobian evaluations, LU-decompositions, forward and backward substitutions and relative computational times. Solver tolerances RELTOL = ABSTOL = $10^{-6}$. 
Figure 8. Error of the solution versus total numerical linear algebra task.
Development, Implementation and Validation of a Hard Real-Time Multibody Simulation for High-Fidelity Steering Wheel Force Feedback

Roland Pastorino#, Michiel Desloovere#, Fredriek Vanneste#, Pascal Degezelle†, Wim Desmet#

PMA division
Department of Mechanical Engineering, KU Leuven
Celestijnenlaan 300b, 3001 Leuven, Belgium
(roland.pastorino, wim.desmet)@mech.kuleuven.be

Laboratorio de Ingeniería Mecánica
Universidad de La Coruña
c/ Mendizábal s/n, 15403 Ferrol, Spain
rpastorino@udc.es

Optidrive N.V.
Wijgmaalsesteenweg 108, 3012 Leuven (Wilsele), Belgium
pascal.degezelle@optidrive.be

ABSTRACT
The development, implementation and validation of a hard real-time forward-dynamics multibody simulation for high-fidelity steering wheel force feedback is presented. A 3D multibody model of a mechanical steering system has been modeled in mixed cartesian coordinates. The equations of motion are solved using an augmented Lagrangian formulation of index-3 with mass-orthogonal projections of velocities and accelerations, together with the generalized-α method and Newmark as time stepping method. Automatic programming techniques including source code generation and source code translation have been used to generate lean and computationally-efficient source code for the multibody model. Hard real-time has been achieved using the EtherCAT fieldbus system that guarantees the determinism of the real-time tasks. The steering wheel torque has been retrieved from the corresponding constraint reaction force. The EtherCAT fieldbus system has been used to interface, with minimum latencies and maximum update rate, the control system running the model with sensors and actuators. The highest steering wheel frequency of interest has been determined based on the haptic capabilities of the human body. Validations on a test bench show that the developed multibody simulation together with its control system are suitable for hardware-in-the-loop simulators. Therefore, accurate and timely steering wheel force feedback is provided to the driver.

Keywords: multibody dynamics, hard real-time, steering force feedback, haptics, hardware-in-the-loop, human-in-the-loop, Xenomai, EtherCAT

1 MODEL-BASED HARDWARE- AND HUMAN-IN-THE-LOOP SIMULATION
1.1 Introduction
Hard real-time simulations of mechanical models are the foundation of advanced Hardware- and Human-In-The-Loop driving simulators. In such simulators, a numerical simulation of the vehicle and/or its sub-systems interacts continuously with the analog environment (including human and hardware) through sensors and actuators. This interaction must be made at regular time intervals and with minimal time delays. A loss of time synchronism causes delays in the sensor data sampling and in the actuator control, resulting in simulation precision losses as well as control instabilities. A prerequisite to developing such an application is the hard real-time simulation of the system model. Consequently, several requirements have to be fulfilled. First the simulation algorithms as well as their numerical implementation must be computationally efficient.
To achieve this when simulating on CPUs, it is crucial to a) choose an efficient formulation of the equations of motion, b) select a stable and accurate integrator, c) implement efficient matrix computations, d) take advantage of the hardware and compiler features that can speed-up the computations, e) select the electronic hardware (CPU, RAM memory, etc) and configure it in order to avoid bottlenecks, f) select an efficient operating system (if any). In addition, the simulation as well as the electronic hardware used in the simulator must have a deterministic behavior. For these reasons it is also essential to a) choose a fixed time step integrator, b) limit the maximum number of iterations if an implicit integrator is used, c) avoid any non-deterministic programming techniques, d) use a Real-Time Operating System (RTOS) (if any), e) sample sensor data and control actuators deterministically and f) use deterministic real-time network communication protocols.

1.2 State-of-the-art of models for steering wheel force feedback

The simulation of the steering system of a vehicle together with the tire behavior gives the possibility to provide accurate steering wheel force feedback to the driver. This is of particular interest for applications such as Steer-by-Wire (SbW) systems and high-fidelity driving simulators. Firstly, vehicles equipped with SbW systems have no mechanical linkage between the steering wheel and the vehicle wheels. One of the difficulties related to this technology is to replicate the steering wheel force feedback. To achieve this, researchers have developed models (an important part being analytical models) of the steering system that calculate the desired steering wheel feedback torque [1, 2, 3]. Secondly, models for high fidelity driving simulators are commonly based on advanced real-time multibody (MB) models [4], which are particularly well suited to accurately simulate the dynamics of the vehicle and its sub-systems. Few such systems based on MB analysis have been developed, e.g. the National Advanced Driving Simulator (NADS). The steering system model of the NADS is a multibody model including a power assist system and a dead band for realistic on-center feeling. A third-order Adams-Bashford has been used as integrator [5]. Shiiba and Suda have developed a driving simulator based on a full vehicle multibody model consisting of 13 rigid bodies [6]. They treat the Jacobian matrix as a constant matrix to reduce the computational effort of solving the equations of motion. A time step of 2 ms has been reached using a Runge-Kutta integrator. Shiiba and Murata have investigated the influence of the so-called "mass-less links", which are essentially rigid bodies replaced by a distance constraint, in a full vehicle multibody model in order to drastically reduce the computation time.[7].

1.3 Developed steering wheel force feedback system

A driving simulator with steering wheel force feedback has been built (see Figure 1). It consists of a steering wheel equipped with an AC permanent magnet synchronous direct-drive motor, its corresponding electric drive, a real-time control system, an Ethernet-based fieldbus system, a chassis, a seat and a virtual-reality headset. The overall architecture of the simulator is presented in Figure 3. The execution flow is explained hereafter. Firstly, the steering wheel angle is acquired and sent to the real-time control system through the Ethernet-based fieldbus. Then, a Kalman filter, running in a real-time task on the control system (see Figure 3), estimates the instantaneous angular position, velocity and acceleration of the steering wheel using the encoder as sensor. The steering wheel position, velocity and acceleration are fed into a multibody model of a mechanical steering system (see Figure 2), which includes a tire model, through a kinematical guidance. The multibody model, executed in a second real-time task, computes one evaluation of the equations of motion (see Figure 3). The computed steering wheel torque is retrieved via the constraint reaction force associated with the Lagrange multiplier of the kinematical guidance constraint. This torque is then sent back to the electric drive through the Ethernet-based fieldbus and finally applied to the steering wheel by the direct-drive motor (controlled in torque) thus closing the control loop.

This paper describes the development, implementation and validation of the hard real-time multibody simulation as well as the hardware-related aspects for this high-fidelity steering wheel force feedback system. The steering wheel force feedback system design is presented in Section 2.
System requirements based on the haptic feedback characteristics of interest have been defined in order to guide the selection of the electric motor. All the aspects related to computationally-efficient and real-time multibody modeling are presented in Section 3 including the multibody formulation, the integrator, automatic source code generation and translation. Section 4 explains the aspects related to hard real-time processing, hard real-time interfacing, and shows the computational performance evaluation. The performance of the force feedback steering system is evaluated in Section 5. Finally, conclusions are drawn in Section 6.

2 STEERING WHEEL FORCE FEEDBACK SYSTEM DESIGN

2.1 Haptic system requirements

The research field on haptic feedback has a broad range of applications such as surgical robots, virtual prototyping for assembly and maintenance assessment and more recently in automotive safety features. Klazky and Lederman divide the sense of touch into cutaneous, kinesthetic and haptic systems [8]. Based on neural inputs, the cutaneous system engages receptors in the skin while receptors located in the muscles, tendons and joints are engaged for the kinesthetic systems. Both are combined in the haptic sensory system. However, the latter is associated with an active behavior, meaning that it is combined with a controlled body motion. Exploring the surface of a steering wheel is an example of an active cutaneous system. Manipulating the steering wheel and feeling the feedback is an example of kinesthetic touch. This research focuses on the second aspect. Computing and generating the forces needed for user interaction with a virtual environment is called haptic rendering [9]. This type of feedback is particularly important in high-fidelity driving simulators in order to provide accurate information to the driver. Haptic rendering is challenging due to the high update rates, which result in a high computational cost and require low latencies. The two main rendering tasks are the computation of the 3D positions and orientations of the virtual objects and the computation of the forces and torques for feedback to the user. A haptic system, such as a steering wheel force feedback system, consists of a human user (i.e. the driver), a computing device (i.e. the real-time control system), the haptic device (i.e. the the steering wheel and its motor) and a virtual environment(i.e. the multibody simulation). To achieve a stable high-fidelity dynamic system, it is essential to have a high update rate for the rendering. Previous research shows that users can distinguish between simulation update rates such as 500Hz and 1kHz [9]. It is important to note that the bandwidth of active movements for humans is much lower then these update frequencies. In fact, the sensing bandwidth ranges approximately from 20 to 30 Hz, the tactile bandwidth is 400Hz and the roughness perception has a bandwith of 5 to 10 kHz. Finally, research applied to steering wheel vibrations [10] shows that the vibrational energy delivered by the steering wheel can reach frequencies of up to 300Hz and resonances of the steering wheel and column are situated between 20 and 30Hz [10]. The aforementioned frequency bands are key to define the requirements of the HIL system. In this research, the target frequency of the HIL system is 30 Hz as it corresponds to the sensing bandwidth and includes the lower
resonances of the steering wheel and column. The simulations in this research have an update rate between 250 Hz and 1kHz.

2.2 Selection of the force feedback motor

A preliminary study has been carried out to evaluate the performances of geared and direct drive motors in the frequency range 0-30Hz in order to assess whether a gearbox could be used for the steering wheel force feedback motor. The test included on the one hand a direct drive Omron R88M-W45015F servomotor and on the other hand an Omron R88M-W40030H servomotor with both a one-stage 5:1 planetary gearbox and a two-stage 15:1 planetary gearbox. The test bench is shown in Figure 8. For the tests, the servomotors were controlled in torque and a swept sine torque profile was applied. The generated torque was measured using a HBM T22 torque transducer and compared to the reference torque to obtain a frequency response function (FRF). The resulting FRFs are displayed in Figure 4. The FRF of an ideal drive-motor-gearbox combination has an amplitude of one and zero phase deviation at all frequencies. The motor with the two-stage gearbox showed a strong decrease in performance with increasing frequency. This performance drop can be attributed to the backlash in the gearbox. Above 20 Hz the backlash effect especially dominated the torque output of the motor. The motor with one-stage gearbox showed only a minor drop in performance compared to the direct drive, but the backlash produced a disturbing rattling noise at higher frequencies that made this solution unsuitable for a driving simulator. Consequently, it was concluded that a direct-drive motor is the preferred solution.

Two different direct drive AC Permanent Magnet Synchronous Motors with their respective electric drives can be used on the simulator. The first one is an Omron R88M-W45015F servomotor with a nominal torque of 2.84 Nm and peak torque of 8.9 Nm. The second is the Omron R88M-K2K030C servomotor with a nominal torque of 6.37 Nm and peak torque of 19.1 Nm.
3 EFFICIENT MULTIBODY DYNAMICS FOR REAL-TIME APPLICATIONS

3.1 Multibody model of the steering system and tire model

A 3D rigid-body model of a real mechanical steering system has been developed (see its graphical representation in Figure 2). It includes the steering wheel, both steering column shafts, the rack and pinion system, the tie rods and the knuckles. The model parameters (masses, inertia, geometry) have been measured and identified experimentally. The system has been modeled using 26 dependent coordinates: 21 natural coordinates and 5 relative coordinates to facilitate the model design. These are summarized in Table 1. These dependent coordinates have been constrained by 25 constraint equations thus resulting in a model with 1 Degree-Of-Freedom (DOF). This DOF (i.e. the angle of the steering wheel) has then been kinematically guided using filtered encoder data. The tire behavior has been modeled following the bore torque model of the TM easy tire model [11].

3.2 Multibody formulation and integrator

Since the early days of the simulation of multibody systems on computers, the development of efficient combinations of integrators and formulations of the equations of motion has been a research topic in the community [12]. As a consequence, a multitude of formulations and integrators have been used over the past 20 years for real-time simulations. In this research, the following criteria have been used to select the formulation and the integrator: a) the simulation has to be stable for a time step in the range of 0.5 to 10 ms. b) it should be possible to add numerical damping in order to stabilize the simulation for large time steps and when sensor noise is present c) the forces and torques corresponding to the constraints should be easily calculated. Following these requirements, an index-3 augmented Lagrangian formulations with projections of velocities and accelerations has been selected together with the generalized-α integrator and a fixed time step [13]. It is explained hereafter in greater detail.

Let us consider a multibody model whose configuration is defined by a set $q$ of $n$ coordinates ($q \in \mathbb{R}^n$) related by a set $\Phi$ of $m$ holonomic constraints ($\Phi \in \mathbb{R}^m$). The equations of motion in Lagrangian coordinates, shown in Equation (1), constitute a set of $n + m$ mixed differential algebraic equations.

$$\dot{M} \ddot{q} + \Phi^T \lambda = Q$$

(1a)

$$\Phi(q,t) = 0$$

(1b)

where $M = M(q) \in \mathbb{R}^{n \times n}$ is the mass matrix, $\ddot{q} \in \mathbb{R}^n$ is the acceleration vector, $Q = Q(q, \dot{q}) \in \mathbb{R}^n$ is

![Figure 4: FRFs for the direct drive, one-stage and two-stage gearbox motors](image)

Table 1: Elements of the multibody model.

<table>
<thead>
<tr>
<th>Element</th>
<th>Point</th>
<th>Vector</th>
<th>Angle</th>
<th>Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>1</td>
</tr>
</tbody>
</table>

![Graph](image)
the vector that contains the external and non-conservative forces as well as the velocity-dependent inertia forces, \( \dot{q} \in \mathbb{R}^n \) is the velocity vector, \( \Phi = \Phi(q, t) \in \mathbb{R}^m \) is the vector of constraints, \( \Phi_q \in \mathbb{R}^{m \times n} \) is the Jacobian of the constraint equations, and \( \lambda \in \mathbb{R}^m \) is the vector of Lagrange multipliers associated to the constraints. Equation (1) has been solved using the Augmented Lagrangian formulation of Index-3 with projections of velocities and accelerations (i.e. AL13-P) presented in [14, 15]. The corresponding equations of motion are shown in Equation (2).

\[
\begin{align}
M \dddot{q} + \Phi_q^T \alpha \Phi + \Phi_q^T \lambda^* &= 0 \\
\lambda^{* (i+1)} &= \lambda^{* (i)} + \alpha \Phi^{i+1}
\end{align}
\]

where \( i \) is the iteration index of the approximate Lagrange multipliers and \( \alpha \) is the matrix of penalty factors. As mentioned before, the generalized-\( \alpha \) method has been chosen in order to add controlled high-frequency numerical damping. This extra damping is important in the case of HIL and HITL applications for two main reasons. The first one is to improve the stability of MB formulations stated in dependent coordinates and based on penalty techniques [16]. The second reason is to filter the high frequency noise introduced by the sensors and their signal conditioning via the kinematical guidances (such as the noise of the steering wheel encoder). On the one hand, this filtering avoids the noise propagation through the equations of motion, the constraint reaction forces and most importantly through the force feedback system. On the other hand, it allows the Kalman filter mentioned in Section 3.4 to introduce less delay at the expense of increased noise. The generalized-\( \alpha \) method [17] transforms the equations of motion shown in Equation (2) as follows:

\[
\begin{align}
(1 - \delta_m) M \dddot{q}_{k+1}^{(i)} + \delta_m M \dddot{q}_k &+ (1 - \delta_f) (\Phi_q^T \lambda_{k+1}^{* (i)}) + \delta_f (\Phi_q^T \lambda_k^*) + \\
(1 - \delta_f) (\Phi_q^T \alpha \Phi) + \delta_f (\Phi_q^T \alpha \Phi) &= (1 - \delta_f) Q_{k+1}^{(i)} + \delta_f Q_k
\end{align}
\]

where \( k \) is the time-step index, \( \delta_m \) and \( \delta_f \) are scalar parameters of the generalized-\( \alpha \) method, \( \lambda_{k+1}^0 = \lambda_k^* \). The time stepping equations for the integration are the Newmark equations, which are shown in Equation (4).

\[
\begin{align}
\dot{q}_{n+1} &= \frac{\gamma}{\beta \Delta t} q_{n+1} + \left[ \frac{\gamma}{\beta} - 1 \right] q_n + \left( \frac{\gamma}{2 \beta} - 1 \right) \Delta t \dot{q}_n \\
\ddot{q}_{n+1} &= \frac{1}{\beta \Delta t^2} q_{n+1} - \frac{1}{\beta \Delta t} \dot{q}_n + \left( \frac{1}{2 \beta} - 1 \right) \ddot{q}_n
\end{align}
\]

where \( \Delta t \) is the time step, \( \beta \) and \( \gamma \) are scalar parameters.

### 3.3 Calculation of the constraint reaction for the steering wheel torque

For the driver force feedback, the steering wheel torque is retrieved from the constraint reaction force associated with the Lagrange multiplier of the kinematical guidance constraint. It has to be noted that, for the formulation AL13-P, it is particularly important to also take into account the constraint reaction coming from the projection stage [13]. Therefore, the steering torque \( \tau_{sw} \) is composed of the constraint reaction from the dynamic equilibrium stage associated to the kinematical guidance of the steering wheel \( \tau_{sw}^{dyn} \) and the constraint reaction coming from the acceleration projection stage \( \tau_{sw}^{acc} \), as shown in Equation (5).

\[
\tau_{sw} = \tau_{sw}^{dyn} + \tau_{sw}^{acc}
\]

\( \tau_{sw}^{dyn} \) can be calculated using the following expression obtained from Equation (3).

\[
\tau_{sw}^{dyn} = -(1 - \delta_f) \left[ \Phi_{sw}^T \lambda_{k+1}^{* (i)} + \alpha_{sw} \Phi_{sw} \right] - \delta_f \left[ \Phi_{sw}^T \lambda_k^* + \alpha_{sw} \Phi_{sw} \right]
\]

169
\( \tau^{acc}_{sw} \) can be calculated using the following expression obtained from TBD.

\[
\tau^{acc}_{sw} = - \left( (\Phi_{sw})_{\eta}^{T} \right) \left( \alpha_{sw} \Phi + k_{sw} \right)
\]  

(7)

3.4 Filtering of sensor data for kinematical guidance

The encoder data has been fed into a Kalman filter in order to estimate the instantaneous position, velocity and acceleration of the steering wheel, which are necessary for the kinematical guidance. The corresponding code is executed in the estimation task (see Figure 3). The filtering introduces inevitably a time delay between the physical and the filtered position, velocity and acceleration signals. From a control point of view, this lag is unwanted because it may cause instabilities. Additionally it may hinder the immersion of the user in the simulation [18]. The amount of delay introduced depends on the filter gains, which are chosen as a function of the sensor noise, and on the time step of the estimation task. The lag introduced on the estimated acceleration is larger than the lag on the velocity, which is in turn larger than the lag on the position estimation. The former is therefore the most critical. For a smaller time step, the predictions of the states become more accurate, resulting in a smaller residual. This means that less corrections are needed in the correction step of the filter and that the estimated quantities react faster to new sensor information. With the current control system, the time step of the estimation task can be reduced up to 100 \( \mu \)s. An offline analysis of the estimator with a 4 Hz sine wave, which is the maximum frequency a human driver can deliver to the steering wheel [19], indicates a lag of 6.4 ms on the estimated acceleration for a 100 \( \mu \)s time step and a lag close of 0.1 ms on the estimated position.

4 HARD REAL-TIME PROCESSING, INTERFACING AND COMPUTATIONAL PERFORMANCE

4.1 Real-time control system

Real-time processing unit - The simulations were executed on a real-time control system, which consists of a personal computer running both Xenomai and GNU/Linux. The reasons for selecting this operating systems are explained hereafter. Two approaches exist to improve the real-time performance of GNU/Linux [20, 21]. The first approach is to patch the kernel in order to make it more preemptible thus improving interrupt latency, context switching time, scheduling latency and providing a finer timer granularity [22]. The second approach is to use a sub-kernel. In the latter, a small real-time kernel runs the Linux kernel as a low priority task. In this research, the second approach was followed, by installing Xenomai 2.6, as it provides better real-time characteristics than the Preempt_RT patch. Xenomai is a minimalist and deterministic operating system that runs aside Linux and on top of Adeos (see Figure 3). Having a higher priority than Linux, its tasks are scheduled first. However, the Xenomai tasks can still switch mode to be scheduled by the Linux kernel thus providing a seamless integration with GNU/Linux operating systems. This seamless integration is key to be able to schedule together real-time tasks (i.e. Xenomai tasks) and non real-time tasks (i.e. Linux tasks). The latter allows easy integration of real-time graphics, graphical user interfaces, data logging, etc. in HIL systems.

Figure 3 shows the software and hardware architecture of the simulator. The processor of the real-time control system is an i7-3740QM whose frequency has been fixed to 2.7 GHz and power-management options disabled. Two periodic real-time tasks of identical priority have been created. These tasks run in hard real-time as they are scheduled exclusively by the Xenomai scheduler (i.e. they never switch mode to Linux). The first one, the estimation task, performs the estimation of the position, velocity and acceleration of the steering wheel at high-frequency (up to 10 kHz) based on the encoder data. The second one, the simulation task, execute the MB model code at a lower frequency (i.e. 250Hz-1kHz). Three periodic non real-time tasks are scheduled by the Linux scheduler. They handle the 3D graphical representation of the MB model, the graphical user interface and the data logging. These five tasks exchange data via shared memory and mutexes.
All pages were locked into memory in order to avoid page faults, which would compromise the real-time performance.

**Sensor and actuator interfacing through Real-Time Ethernet** - The interfacing of the sensors and actuators to the real-time control system has been setup via EtherCAT (Ethernet for Control Automation Technology). This Ethernet-based fieldbus system is suitable for hard real-time interfacing due to its low communication jitter (<1 µs). Other features, which are essential in the scope of this research, include its short cycle times (<100 µs) and its high throughput, thanks to being based on the standard IEEE 802.3 Ethernet frame. The architecture of the interfacing via EtherCAT is shown in Figure 3. The real-time control system is connected to a bus coupler which is in turn connected to the electric drive. The EtherCAT master of the control system has been setup using the IgH EtherCAT master. An EtherCAT-capable Ethernet driver reads and writes the frames from/to memory via Direct Memory Access. Finally, the EtherCAT master communicates with the Xenomai estimation task through the Real-Time Driver Model (RTDM). Following this setup, the EtherCAT communication is entirely handled in Xenomai thus ensuring that real-time performance is preserved for both the numerical computations and the I/O processing.

### 4.2 Automatic source code generation and translation

Automatic programming in the context of real-time simulations of multibody dynamics has been discussed in [23]. The techniques and software developed in this previous research have been used here to speed up the development of the MB model of the steering system. The source code generator gives the possibility to select a) the target language b) the numerical library for linear algebra c) the programming paradigm (e.g. functional programming, object-oriented programming) d) the MB formulation e) the integrator f) the compiler options g) and the operating system. It is an asset in the development of HIL/HITL applications [23]. In this work, automatic programming has proven to be a valuable tool. Source code has first been generated in Python together with the library Numpy, for rapid code prototyping during the creation of the definition of the MB model. Once this definition was completed, lean procedural C++ source code has been automatically generated for the real-time simulation of the MB model on the aforementioned real-time control system, thus saving programming and debugging efforts. The C++ linear algebra library Armadillo has been used for efficient numerical computations. The standalone generated code for the model therefore requires a minimal amount of RAM, takes full advantage of the compiler optimization due to its reduced size and simplicity, and can easily be integrated with other software libraries for hard real-time network communication, 3D graphics, graphical user interface, etc.

### 4.3 Computational performance evaluation

In order to measure the computational performance of the generated MB model, lean code has been generated together with extra code for measuring and logging every integration step duration. It has to be noted that, for this test, the simulation has been executed in real-time for 60 minutes. Therefore, following the aforementioned hard real-time requirements, the integration duration must always be shorter than the time step duration. The histogram of integration step durations is shown in Figure 5 and the corresponding statistics are summarized in Table 2. The maximum integration duration is well below the time step duration (i.e. 1 ms) thus confirming the hard real-time behavior.

### 4.4 Real-time performance evaluation

Section 4.2 dealt with software techniques for generating efficient code for the MB model. The computational performance of the generated code was discussed in Section 4.3. The computational efficiency is a necessary but not sufficient condition to simulate MB models in real-time. Numerous definitions of real-time system can be found in the literature [24]. The one taken into
account in this research is: “A real-time system is one whose logical correctness is based on both the correctness of the outputs and their timeliness”. It is worth mentioning that in the field of MB systems, the word real-time is mainly used to refer to efficient computational methods instead of being related to the deterministic response of the system or simulation.

Real-time computing is broadly and subjectively divided into soft and hard real-time computing. In the first case, the timeliness of the computations is essential but not critical. A common example is the simulation of a real-time MB simulation on a vanilla Linux kernel with, for example, a 5 ms time step. In such a situation, the jitter of this operating system and the time step will be of the same order of magnitude. Whether this is an acceptable situation or not depends on the requirements of the application. On the other hand, hard real-time simulation are executed at regular time interval in a deterministic manner. The timeliness of the computations is critical. Therefore, in order to run a real-time MB simulation, time determinism and low latencies are mandatory as well as computational efficiency. This research focuses only on the hard real-time aspects.

The real-time performance of real-time operating systems like Xen may be negatively affected by its system load. In the case of MB simulations, the main system load comes from the numerical computations. Subsequently it is important to compare the real-time performance with and without load. Tests have been performed for the 3D steering system by generating lean Xen-enabled C++ code. All tests were executed for 60 minutes. Figure 7 shows the probability distribution for a 1 ms time step duration in the cases where no system load exists and where the MB model loads the system. The statistics related to these measurements are summarised in Table 2. Results show that the maximum error for the time step duration in both test cases is lower than 45 µs. The standard deviation of the time step durations is only slightly bigger in the case in which the system is loaded. It can therefore be concluded that the MB simulation does not compromise the determinism of the real-time tasks.

5 PERFORMANCE EVALUATION OF THE HIL SYSTEM

The performance of the HIL system has been evaluated on the dedicated test rig shown in Figure 8, in which an electric motor replaces the steering wheel and the driver. A torque sensor has been placed in between the steering motor and the force feedback motor. The former is controlled in
Figure 7: Real-time performance with and without system load on Xencma i

Figure 8: Test setup for HIL system performance evaluation

position and the latter in torque. The torque resulting from the behavior of both the MB model and the tire model at low frequencies is shown in Figure 6 using a sine wave of frequency 0.2Hz and amplitude 90° as steering wheel position reference.

Latency measurements - The maximum latency of the whole control loop (including the MB model) is one of the most important specifications of a HIL/HITL system. For this reason, a test has been prepared to measure the maximum latency between the acquisition of an encoder value and the torque response corresponding to this new position. For such a test, a periodic discontinuity has to be introduced in the system and measured at the output, during an extended period of time. It is not practical to perform this test using the torque computed by the MB model as it includes many nonlinearities that complicate the identification of the time delays. Therefore, the function shown in Equation (8) has been used, in the simulation real-time task, to generate the output torque. It has to be noted that the MB model is still executed in this task.

\[
\tau_{sw} = \begin{cases} 
  h & \text{if } \theta \geq 0 \\
  -h & \text{if } \theta < 0 
\end{cases}
\]  

(8)

where \( h \) is the torque amplitude and \( \theta \) is the steering wheel angle. Figures 9a and 9b show the distribution of latencies for a simulation time step \( \Delta t \) of 1 ms and 5 ms, and a logging time step of 200 \( \mu s \). For both figures, the maximum delay is equal to the simulation time step plus a constant delay of 2.2 ms.

Selection of the time steps of the real-time tasks - The performance of the developed HIL system can be tuned by changing the time steps of the different real-time tasks. The time step
of the estimation task controls the delay and the noise introduced by the Kalman filter for the kinematical guidance while the one of the simulation task controls the maximum frequency that the MB model can represent as well as the maximum latency of the HIL system (at least for an important part). In this research, the frequency range of interest is 0-30Hz. Therefore, for example, a simulation time step of 1 ms would give a delay of 3.2 ms which corresponds to about 10% of the minimum time period (i.e. 33.3 ms).

6 CONCLUSIONS

This research has shown the development, implementation and validation of a hard real-time forward-dynamics multibody simulation of a mechanical steering system for high-fidelity steering wheel force feedback. The requirements of the hardware-in-the-loop system including those of the multibody model, the sensor and actuator interfacing, have been established based on haptic characteristics. Then, the 3D multibody model has been modeled in mixed cartesian coordinates and the equations of motion solved using an augmented Lagrangian formulation of index-3 with mass-orthogonal projections of velocities and accelerations, together with the generalized-α method and Newmark as time stepping method. Automatic programming methods including source code generation and source code translation have been used to generate lean and computationally-efficient source code for the multibody model. The steering wheel torque has been retrieved from the corresponding constraint reaction force. The Xenomai real-time operating system guarantees the required determinism for the real-time tasks while the EtherCAT fieldbus system has been used to interface, with minimum latencies and maximum update rate, the control system running the model with sensors and actuators. Validations on a test bench have shown that the developed multibody simulation and its control system can give high-fidelity driver steering force feedback up to 30 Hz. Further work could extend this research by including a model of the hydraulic power assist pump in addition to the MB model via coupled simulation or co-simulation. A full-vehicle MB model with a detailed 3D environment would also allow to provide a realistic haptic feedback to the driver based on the road conditions.

Acknowledgments

The authors gratefully acknowledged the support of a) the IWT Flanders within the MBSE and MODRIO projects b) the Research Fund KU Leuven c) ITEA2 through the MODRIO project d) Flanders Make strategic research centre and e) the Belgian Programme on Interuniversity Attraction Poles, initiated by the Belgian Federal Science Policy Office (DYSCO)

References


Enhancing the Performance of the DCA When Forming and Solving the Equations of Motion for Multibody Systems

Jeremy J. Laflin*, Kurt S. Anderson*, Michael Hans*

* Rensselaer Polytechnic Institute
Troy, NY 12180, USA
laflij@rpi.edu, anderk5@rpi.edu, hansm@rpi.edu

ABSTRACT
This paper provides an initial investigation into using the Graphics Processing Unit (GPU) (or similar hardware) to execute the Divide-and-Conquer Algorithm, which forms and solves the equations of motion for articulated multibody systems. The computational time required to form and solve the equations-of-motion of a simple $n$-length pendulum using the GPU is compared with a standard serial CPU implementation, a rudimentary parallelization on the CPU using OpenMP, and some combinations of the CPU and the GPU. The hybrid version uses the GPU for a select number of levels in the recursive sweeps and uses an OpenMP parallelization on a multi-core CPU for the remaining levels of recursion.

The results demonstrate a significant performance increase when the GPU is used despite recursive algorithms being ill-suited to hardware designed for Single Instruction Multi-Data (SIMD). This is largely due to the tree-type structure of recursive processes, with half of the required operations being contained in the first level of recursion for a binary tree.

1 INTRODUCTION
Since computational performance is critically important for simulations to be used as an effective tool to study and design dynamic systems, the computing performance gains offered by GPUs should not be ignored. The GPU has been used to increase the computational performance of many tasks necessary to simulate multibody systems [6, 10, 11, 15, 16]. Since the GPU is designed to execute a very large number of simultaneous tasks (nominally SIMD), recursive algorithms in general, such as the DCA, are not well suited to be executed on GPU-type architecture. This is because each level of recursion is dependent on the previous level. Therefore, all tasks associated with the algorithm cannot be executed independently. The primary issue is the amount of data transfer that must occur when moving from one level of recursion to the next. However, the GPU can be leveraged to increase computational performance when using the DCA to form and solve the equations of motion for articulated multibody systems with a large number of degrees-of-freedom due to the tree structure of DCA.

Computational performance of dynamic simulations is highly dependent on the nature of the underlying formulation and the number of generalized coordinates used to characterize the system. Therefore, algorithms that scale in a more desirable (lower order) fashion with the number of degrees-of-freedom are generally preferred when dealing with large ($N > 10$) systems. However, the utility of using simulations as a scientific tool is directly related to actual compute time. The DCA, and other top performing methods, have demonstrated the desirable property of the required compute time scaling linearly with ($O(n)$) with the number of degrees-of-freedom ($n$) and sublinearly ($O(\log n)$) when implemented in parallel. However for the DCA, total compute time could be further reduced using parallel hardware, such as the GPU, by exploiting the large number of independent operations involved in the first few levels of recursion.

1.1 The DCA for Forming and Solving Equations of Motion for Multibody Systems
The DCA was first introduced by Featherstone for both open-loop [4] and closed-loop [5] topologies and increases the efficiency with which the equations-of-motion can be formed and solved...
numerically. Additionally, there have been a number of modifications to the original method [1–3, 7–9, 13, 14, 17, 18]. The basic method is reproduced herein using the notation of Mukherjee and Anderson [12].

The DCA consists of two recursive processes: assembly and disassembly. These recursive processes take place using a hierarchical tree structure. The tree structure is defined by the kinematic joints connecting the bodies of the system. Typically the inboard and outboard joint of the body coincides with a reference point called a handle. The inverse inertial properties of two adjoining parent bodies are combined to represent a fictitious assembly (child body), see figure. 1. This is possible because the kinematics of the joint are known, which allows the constraint forces acting at the connecting joint to be excluded from the equations-of-motion of the outboard handles of the parent bodies (the handles of the child body). The relative motion between the parent bodies is captured by an equation that describes the amount of motion happening in the directions of motion that are allowed by the joint, which are known. This process is repeated until there is only one body, the root body (see Fig. 2). At this point the boundary conditions are known and the equations-of-motion can be solved, which do not contain any of the constraint forces acting at the non-terminal joints.

In general, a body of the system may possess any number of handles, though the basic aspects
of the method are most easily conveyed by discussing a chain system. For such systems, each body possesses two handles that connect the body to its inboard and outboard neighbors. The basic method involves writing the spatial equations-of-motion corresponding to these two reference points (handles) \( H^i \) \( i = 1, 2 \) on each body, as

\[
A^k_i = \zeta^k_{ij} F^k_{1c} + \zeta^k_{ij} F^k_{2c} + \zeta^k_{ij}
\]

(1)

\[
A^k_2 = \zeta^k_{ij} F^k_{1c} + \zeta^k_{ij} F^k_{2c} + \zeta^k_{ij}
\]

(2)

and

\[
A^{k+1}_1 = \zeta^{k+1}_{ij} F^{k+1}_{1c} + \zeta^{k+1}_{ij} F^{k+1}_{2c} + \zeta^{k+1}_{ij}
\]

(3)

\[
A^{k+1}_2 = \zeta^{k+1}_{ij} F^{k+1}_{1c} + \zeta^{k+1}_{ij} F^{k+1}_{2c} + \zeta^{k+1}_{ij}
\]

(4)

These bodies, \( Body^i \) and \( Body^{k+1} \), connected by a kinematic joint \( j \) and therefore are subject to the kinematic constraint

\[
P^i \dot{u} = A^{k+1}_1 - A^k_2 - \dot{P}^i u.
\]

(5)

\( A^k_i \), and \( F^k_i \), are the spatial acceleration of, and force on, handle \( i \) respectively and are defined as

\[
A^k_i = \begin{bmatrix} \ddot{\alpha}^k_i \\ \dddot{a}^k_i \end{bmatrix}
\]

(6)

and

\[
F^k_i = \begin{bmatrix} \tau^k_i \\ f^k_i \end{bmatrix}
\]

(7)

The rotational acceleration of \( Body^i \) is \( \ddot{\alpha}^i \), \( \dddot{a}^i \) is the translational acceleration of the reference point \( H^i \), and \( \overrightarrow{\alpha}^i \) and \( \overrightarrow{f}^i \) are the constraint torques and forces acting at \( H^i \) respectively. The \( \zeta^k_{ij} \) \( i, j = 1, 2 \) terms are the spatial matrix representations of the inverse inertial properties at the handles, while \( \zeta^{k+1}_{ij} \) \( i = 1, 2 \) contains applied forces acting on the body and other velocity dependent terms. The resulting set of equations, Eqns. (1 - 4), can be reduced by exploiting the fact that the constraint forces are equal and opposite, i.e, \( F^k_2 = -F^{k+1}_1 \), and that the kinematics of the connecting joint are specified. Specifically, Eqn. (5) describes the relative acceleration between connecting bodies using the generalized acceleration \( \dot{u} \) along known directions defined by the connecting joint partial velocity (mode of motion) \( P^i \). The equations-of-motion for the assembled fictitious pseudo-body \( Body^{k+1} \), at \( H^k \) and \( H^{k+1} \) can be expressed as

\[
A^k_1 = \zeta^{k+1}_{ij} F^{k+1}_{1c} + \zeta^{k+1}_{ij} F^{k+1}_{2c} + \zeta^{k+1}_{ij}
\]

(8)

and

\[
A^{k+1}_2 = \zeta^{k+1}_{ij} F^{k+1}_{1c} + \zeta^{k+1}_{ij} F^{k+1}_{2c} + \zeta^{k+1}_{ij}
\]

(9)

by algebraically eliminating the constraint forces at the connecting joint. The resulting equations (8 & 9) are of the same form as the equations-of-motion for the handles of any generic body.

In the above equations, \( \zeta^{k+1}_{ij} \) represents the inertial quantities of the fictitious pseudo-body resulting from the assembly of \( Body^i \) and \( Body^{k+1} \). For the derivation of the inverse inertial terms and the details of the assembly process, the reader is referred to the work of Featherstone [4] or Mukherjee and Anderson [12]. This assembly process is then repeated recursively, until only a single assembled pseudo-body remains (root body), as shown in Fig. 2. This is possible because the form of the equations-of-motion for the handles of an assembled body is indistinguishable from the form of the equations-of-motion for the handles of a generic body. The assembly process yields the equations-of-motion associated with the two boundary handles

\[
A^1_1 = \zeta^{1,n}_{11} F^{1,n}_{1c} + \zeta^{1,n}_{12} F^{1,n}_{2c} + \zeta^{1,n}_{13}
\]

(10)
and

\[ A_1^2 = \zeta_{21}^1 F_{1c}^1 + \zeta_{22}^1 F_{2c}^2 + \zeta_{23}^1, \]  

(11)

which are written in terms of only the spatial inertial quantities of all bodies in the system and the constraint forces acting at the two handles of the root body (boundary handles).

The spatial accelerations of, and constraint forces acting at \( H_1 \) and \( H_2 \) can now be determined using the known boundary conditions. After determining these quantities, the disassembly process begins, in which all unknown spatial accelerations of the handles and constraint forces acting at all connecting joints are determined. This recursive process determines the constraint forces acting at a joint in terms of the constraint forces acting at the handles of the assembly, and the inertial properties of the assembled body, as

\[ F_{k+1} = W_{23}^k F_{1c}^k - W_{12}^k F_{2c}^k + Y. \]  

(12)

The terms \( W \) and \( Y \) are terms containing inertial properties from the assembly of the two bodies, see Featherstone [4], or Mukherjee and Anderson [12] for derivation of these terms. Once this constraint force acting at a joint is determined, the spatial accelerations of the handles that are connected by this joint can be determined using Eqns. (8 & 9). This allows the computation of the generalized acceleration (\( \dot{u} \)) at the joint using Eqn. (5).

1.2 Potential Challenges Executing the DCA on the GPU

Because this algorithm is by nature a recursive one, at first it seems ill-suited to run on the GPU architecture. This is because, by definition recursive algorithms can not be completely parallelized and require a minimum number of sequential operations. These sequential processes become problematic due to the potentially high cost of data transfer from the CPU memory to the GPU device memory. The lower connection speed is due to the physical arrangement of the CPU and the GPU and the type of connections used, which is continually improving.

The recursion becomes problematic, for the DCA, because the number of floating-point operations that need to be performed is low at the levels of recursion approaching the root body. Therefore, there is potentially a large amount of computational overhead in comparison to the number of floating-point operations that must be performed. This is contrary to the purpose for which GPUs were designed, which is to produce a large amount of throughput. For this reason, it may seem that other algorithms which may scale more poorly with the number of bodies in the system, but can be more completely parallelized may achieve a lower compute time than the DCA or other recursive algorithms.

Although data transfer time is a significant source of overhead, there are others that are problematic for recursive algorithms, for example, kernel launch times and synchronization. Kernel launch time is particularly problematic for the DCA because the DCA consists of two operations, assembly and disassembly, which are each separate kernels and must be launched recursively. The recursive nature of the assembly and disassembly processes also necessitates that the kernels are synchronized at each level of recursion adding even more computational overhead.

1.3 Using the GPU Effectively With the DCA

Despite the apparent drawbacks of executing the DCA on the GPU, there are ways the inexpensive and powerful computing resources offered by the GPU can be utilized with the DCA. For large systems, there are a large number of independent assembly and disassembly operations that occur in the levels near the leaf level of the recursive tree. For a system of 2048 bodies the approximate operations counts for each level of recursion are given in Tab. 1. From Tab. 1 it is estimated that 50% of the operations per kernel sweep are performed in the first level and 75% are performed in the first two levels of recursion. Therefore, 50% of the operations per sweep can be parallelized with the penalty of associated kernel launch and no synchronization penalty. Furthermore, 75% of
the operations can be parallelized incurring the data transfer and synchronization penalties of only one level of recursion in addition to the leaf level operations. As the systems size increases, using the GPU to perform the operations in these levels could allow much larger problems to be investigated by effectively halving or quartering the original problem seen by any CPU parallelization strategy.

<table>
<thead>
<tr>
<th>Level</th>
<th>Leaf</th>
<th>Leaf+1</th>
<th>Leaf+2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Bodies</td>
<td>2048</td>
<td>1536</td>
<td>1792</td>
</tr>
<tr>
<td>% Operations / Sweep</td>
<td>50</td>
<td>75</td>
<td>88</td>
</tr>
</tbody>
</table>

Table 1: Estimation of Operations per Level of Recursion of the DCA

By using the GPU in only the levels that have a relatively high number of parallel operations, the computational overhead compared to the number of floating-point operations is kept low. The CPU can then be used only for the remaining levels in which there is a large number of sequential computations in comparison to a relatively low number of floating-point operations per level. In this way the GPU can be used to provide a large amount of throughput outweighing the overall overhead required.

Additionally, there are many ways that the data communication cost can be effectively hidden and the impact on overall simulation time can be minimized. For just one example of this consider a simulation of a bio-molecular system where coarse graining has resulted in an articulated multi-body system. In such an example, the data corresponding to the leaf level bodies can be transferred while computing the electrostatic forces.

Since the impact of the data communication cost associated with approximately 50% of the operations can mitigated, the penalty of using the GPU for two levels of recursion can be reduced to the cost of the data transfer between levels one and two (~ 1/4 of the overall data), the synchronization between the two levels, and the second launch of the associated kernels.

2 IMPLEMENTATION AND NUMERICAL EXAMPLE

A simple numerical example was performed to investigate the computational advantages of executing the DCA on GPU architecture. The equations of motion are formed and solved using the DCA in a variety of parallel arrangements. The DCA is implemented in serial and parallel on multi-core CPU using OpenMP to parallelize the assembly and disassembly operations happening at each level. Other lower level CPU parallelization implementations are possible and with more effort may yield faster results. However, as a platform to investigate the scalability of the algorithm on the GPU as compared to a parallel CPU implementation, OpenMP is sufficient. The CPU was a Intel® Xeon® W3565 3.20GHz having four cores.

The number of levels of recursive assembly, and subsequent disassembly was selected a priori and arbitrarily. However, with some simple functions to time the data transfer time and kernel launch time the optimal number of levels for a particular hardware combination could be approximately determined. This was not performed because the information is specific to the hardware arrangement and is of little interest to other users. The levels of the DCA that are performed using the GPU were executed on a NVIDIA® Corporation GF108GL Quadro® 600. Therefore, the NVIDIA® CUDA® platform was used for the parallel implementation of the assembly and disassembly kernels.

2.1 Test Case

A multi-link pendulum, as shown in Fig. 3, was chosen as the test system to investigate how the computational time scales with the number of bodies using various approaches to GPU-parallelization.
Although this example is 2D, all matrices and vectors are “full-sized” as they would be in the 3D case so that the data communication cost is nearly the same.

![Diagram of Pendulum Test Problem](image)

Figure 3: VARIABLE LENGTH PENDULUM TEST PROBLEM

The number of bodies was varied up to 2048. The number of levels of recursion that were executed on the GPU was varied from zero to all levels. For 2048 bodies this corresponds to eleven levels of recursive operations. The pendulum was released from rest at \( q_i = 0 \) for all \( i \), and the simulation duration was a single time-step. The compute-time required to complete the assembly and disassembly sweeps, in which the equations of motion are formed and solved including determination of constraint forces acting at the joints, is plotted versus the number of bodies in the system in Fig. 4.

### 2.2 Results

![DCA Compute Time Graph](image)

Figure 4 demonstrates an decrease in computational time as compared to OpenMP when executing the DCA on GPU architecture, even for a moderately large number of bodies. Additionally, it is easily observed from Fig. 4 that parallelizing the assembly and disassembly operations of the leaf level, provides the largest reduction in compute time for this arrangement of 2048 bodies. It is also clear from Fig. 4 that as the number of bodies increases executing more levels of recursion provides still more decreases in computational time, although at a decreasing rate. It can also be seen from Fig. 4 that executing all levels of recursion outperforms the OpenMP solution near 512 bodies, which would mostly likely be reduced with a more sophisticated NVIDIA\(^\text{®}\) CUDA\(^\text{®}\) implementation of the DCA assembly and disassembly kernels. Conversely, this number may be increased with a more advanced CPU parallelization implementation. Finally, it can be seen in Fig. 4 that the speed-up gained from parallelizing all levels on the GPU is approximately double to that of the OpenMP parallelization for 2048 bodies.
3 CONCLUSIONS

Despite the obvious drawbacks to executing a recursive algorithm on GPU architecture, a significant reduction in compute time was observed when doing so with the DCA. For this implementation the compute time is reduced in half when comparing the OpenMP and the best hybrid solution. This is likely due to the large amount of independent operations that occur in the first levels of recursion. Although a hybrid implementation using parallel processes on both the CPU and the GPU provide the lowest compute times for moderately sized systems, the penalty for using the GPU exclusively becomes less important relatively quickly (∼5 ms for 2048 bodies). Therefore for very large systems, such as those encountered in biomolecular simulations where the number of degrees-of-freedom (n) easily exceed 10⁵, a hybrid approach may not be worthwhile.

This significant reduction in forming and solving the equations of motion for a single time-step likely do not translate into a halving the simulation time of real problem due to other calculations which may be performed, such as the previously mentioned electrostatic forcing term determination for biomolecular systems. It does suggest however, that the desirable solve-time scaling properties of the DCA (and other recursive methods) may be preserved while significantly reducing the actual solve time using relatively inexpensive resources.

REFERENCES


Forward Dynamics of Variable Topology Mechanisms
– The Case of Constraint Activation

Andreas Müller
Institute of Robotics, JKU Johannes Kepler University
Altenbergerstr. 69, 4040 Linz, Austria, a.mueller@jku.at

ABSTRACT
Many mechanical systems exhibit changes in their kinematic topology altering the mobility. Ideal contact is the best known cause, but also stiction and controlled locking of parts of a mechanism lead to topology changes. The latter is becoming an important issue in human-machine interaction. Anticipating the dynamic behavior of variable topology mechanisms requires solving a non-smooth dynamic problem. The core challenge is a physically meaningful transition condition at the topology switching events. Such a condition is presented in this paper. Two versions are reported, one using projected motion equations in terms of redundant coordinates, and another one using the Voronets equations in terms of minimal coordinates. Their computational properties are discussed. Results are shown for joint locking of a 3R mechanisms.

Keywords: Time integration, variable topology, non-smooth dynamics, momentum conservation.

1 Introduction
Variable topology mechanisms (VTM) form a class of mechanisms that can switch between different kinematic topologies, and thus change their kinematic mobility and possibly their DOF. This property is known from kinematotropic mechanisms [1, 14, 18]. The latter transit between motion modes via kinematic singularities while keeping their kinematic topology. VTM on the other hand change their mobility due to constraint switching. This can be caused by (bilateral) contacts or joint locking. In this paper it assumed that, except at the switching events, the constraints are smooth and scleronomic. The VTM are then referred to as quasi-scleronomic VTM.

Topology variations lead to discontinuous system trajectories. This is problematic for the numerical simulation as well as for the solution of the inverse dynamics problem within model-based control schemes. Unilateral contacts in multibody systems are the prevailing source of topology changes, and have been the topic of intensive research and the driving force towards a theory for the non-smooth mechanics. An overview can be found in [2, 10, 9]. In this context the modeling of friction is crucial, and various formulations have been reported, e.g. [8, 11, 16]. Another cause for non-smooth dynamics are discontinuous (bilateral) constraints, i.e. switching between different constraints. These became relevant in the context of molecular systems, where the original model is replaced by lower-dimensional surrogate models. The topology change is merely due to a change in the kinematics without the need to consider further constitutive laws (as in case of frictional contact). The dynamics simulation of such models with variable kinematic topology has been addressed in [7, 12, 13]. A condition for momentum conservation within numerical time stepping schemes was proposed.

Beyond molecular dynamics, there is a recent interest in this type of discontinuous systems. Moreover, it accounts for situations with great significance for robotics, and for human-machine interaction in particular. An emergency stop is an excellent example. In such an event the joint brakes are activated. This does not lead to an immediate halt. Rather the system is initially moving according to the brakes’ sliding friction. The individual joints eventually lock and enter the stiction phase. This occurs at different time instances for the different joints. Now, as part of the safety considerations of the human-machine interaction it is crucial to anticipate the trajectory of a robot in emergency situations.

Topology variations must also be resolved within the inverse dynamics providing the feed-forward term in model-based control schemes. As an example, in [15] a method for obstacle-avoiding control
of serial manipulators was proposed, where a virtual joint is introduced whose motion is in some sense perpendicular to the obstacle. This virtual joint is introduced only when possible collision is detected, thus leads to a variation of the topology.

From a computational point of view, the forward dynamics of VTM with switching constraints requires additional compatibility conditions on the velocity and momentum. Such conditions are presented in the following. The paper introduces such conditions for the general case where multiple constraints are successively activated.

2 Configuration Space of Quasi-Scleronomic VTM

The configuration space (c-space) of a holonomic quasi-scleronomic VTM is the time dependent variety

\[ V_i := h_i^{-1}(q, t) \]  

(1)
defined by a system of quasi-scleronomic geometric constraints \( h(q, t) = 0 \). The latter are piecewise defined constraints of the form

\[ h(q, t) = \begin{cases} 
  h_1(q), & t \in [t_0, t_1) \\
  h_2(q), & t \in [t_1, t_2) \\
  \vdots \\
  h_i(q), & t \in [t_{i-1}, t_i) 
\end{cases} \]  

(2)

where the constraint switching happens at \( t_i \). Each individual constraint \( h_i(q) = 0 \) corresponds to a kinematic topology, and defines a variety \( V_i := h_i^{-1}(q) \), where \( V_i \cap V_j \neq 0 \), which is the c-space of the VTM at topology \( i \). Each c-space \( V_i \) is stratified into smooth manifolds of regular points and subvarieties of singular points. The local DOF at \( q \in V_i \) is the highest dimension of submanifolds passing through that point: \( \delta_{\text{loc}}(q) := \dim_q V_i \). The differential (instantaneous) DOF is \( \delta_{\text{diff}}(q) := \dim_{\text{q}} J_i(q) \), with \( J_i \) being the Jacobian of the constraint mapping \( h_i \). A configuration \( q \) is regular iff \( \delta_{\text{diff}} \) is constant in a neighborhood of \( q \) in \( V \).

Definition 1. A topology change at time \( t_0 \) with configuration \( q_0(t_0) \in V_i \cap V_j \) is called regular iff \( q_0 \) is a regular point of \( V_i \) and of \( V_j \). Otherwise it is called singular.

A regular topology change is thus a change of local DOF without an intermediate drop of the differential DOF. This is necessarily due to contact constraints or (time dependent) locking. Apparently this is not possible for kinematotropic mechanisms that must pass through singularities in order to switch between different-dimensional manifolds.

In this paper a momentum consistent formulation for the forward dynamics is presented for VTM undergoing regular topology changes due to the activation of constraints. Hence the VTM does not encounter a singularity during the topology change. This fact is exploited for the momentum consistent transition condition applicable within time stepping schemes. The method is tailored to the case that additional constraints are activated.

3 Suitable Forms of Motion Equations

3.1 Lagrange Equations of First Kind

In the following a form of the EOM is recalled that will be suited for the derivation of the momentum balance. The starting point are the equations of motions (EOM) of an unconstrained MBS of the form

\[ M(q)\ddot{q} + C(q, \dot{q})\dot{q} + P(q) + Q(q, q, t) = u(q, t). \]

Here \( M \) is the generalized mass matrix, \( C \) represents forces quadratic in velocity, \( P \) potential forces, \( u \) applied generalized forces, and \( Q \) any other generalized forces inherent to the system. The vector \( q \in \mathbb{R}^n \) comprises \( n \) generalized coordinates. It is assumed that \( C \) and \( P \) are smooth mappings.

Assume that the MBS is subjected to a system of \( m \) scleronomic geometric constraints \( h(q) = 0 \) that define the c-space \( V = h^{-1}(0) \). The constraint Jacobian, in the corresponding velocity constraints \( J(q)\dot{q} = 0 \), leads to the Lagrangian EOM

\[ M(q)\ddot{q} + C(q, \dot{q})\dot{q} + P(q) + Q(q, q, t) + J^T(q, t)\lambda = u(q, t) \]  

(3)
The above geometric constraints give rise to the acceleration constraints
\[
J(q)\ddot{q} + J(q, \dot{q}) \dot{q} = 0. \tag{4}
\]
For numerical treatment, frequently (3) is combined with the acceleration constraints to yield the index 1 DAE system
\[
\begin{pmatrix}
M(q) & J^T(q) \\
J(q) & 0
\end{pmatrix}
\begin{pmatrix}
\ddot{q} \\
\dot{\lambda}
\end{pmatrix}
= \begin{pmatrix}
-C(q, q) \ddot{q} - P(q) - Q(q, q, t) + u(t) \\
-J(q, q) \dot{q}
\end{pmatrix}. \tag{5}
\]
The advantage of this formulation is that the coefficient matrix is usually sparse since the constraint Jacobian is usually sparse. The sparsity of the mass matrix depends on how the unconstrained system is modeled. If the latter is modeled in terms of absolute coordinates, \(M\) is diagonal. Otherwise, e.g. using relative coordinates, this is a dense block-triangular matrix. Then the size \((n+m) \times (n+m)\) of the coefficient matrix can become a burden for numerical solution. Moreover, when the number of constraints is not constant, the size of the coefficient matrix changes. A formulation with coefficient matrix of constant dimension is presented in the next section. The formulation in terms of minimal coordinates is recalled in section 3.3.

3.2 Projected Motion Equations in terms of Redundant Coordinates – Gauß’ Principle

A system of EOM for the constrained system in terms of the complete set of the (redundant) coordinates \(q \in \mathbb{V}^n\) can be derived from Gauß’ principle written in the form
\[
\begin{cases}
\frac{1}{2} (\ddot{q} - a)^T M(q) (\ddot{q} - a) \rightarrow \min \\
J(q)^T \ddot{q} + \dot{J}(q) \ddot{q} = 0
\end{cases}
\tag{6}
\]
where \(a = M^{-1}(u - C \ddot{q} - P - Q)\) is the acceleration of the unconstrained system. Assuming non-redundant constraints, the solution \(\ddot{q}\) solving (6) is
\[
\ddot{q} = M^{-1}(q)N_{JM}^T(u(q, t) - C(q, q) \ddot{q} - P(q) - Q(q, q, t)) - J_M^+(q) J(q)^T \dot{q}
\tag{7}
\]
with \(J_M^+ = M^{-1}J^T(JM^{-1}J^*)^{-1}\) being the \(M\)-weighted right pseudoinverse of \(J\), and \(N_{JM} = I - J_M^+ J\) the corresponding projector to the null-space of \(J\). This formulation is free of Lagrange multipliers and furthermore does not require selection of independent (minimal) coordinates, as minimal coordinate formulations for constrained MBS do (next section).

As reported in [4], using \(J\ddot{q} = -J\dot{q}\), the equations (7) can be written
\[
N_{JM}^T(M(q) \ddot{q} + C(q, q) \ddot{q} + P(q) + Q(q, q, t) - u(q, t)) = 0. \tag{8}
\]
The formulation (8) is advantageous since the unaltered EOM of the unconstrained system are simply projected onto the c-space (actually its cotangent space). Moreover, a change in the constraints only affects the null-space projector respectively the pseudoinverse.

**Remark 1.** The motion equations (7) form a system of \(n\) independent ODEs in \(q(t)\) that can be used for forward dynamics analysis. The system (8), on the other hand, consists of \(n\) equations of which only \(n - m\) are independent, since \(\text{rank } N_{JM} = n - m\). It is thus not directly applicable to forward dynamics simulation, but allows for a very efficient inverse dynamics solution applicable for model-based control schemes in redundant coordinates [6]. Nevertheless, the equations (8) together with (4) form a system of \(n + m\) motion equations of which \(n\) are independent.

**Remark 2.** The weighted null-space projector, arising from the solution of (6), respectively the elimination of the Lagrange multipliers, ensures the dynamic consistency of the projection in (7). Whereas the image space of \(N_{JM}^T\), i.e. the range space of \(N_{JM}\), depends on the weight \(M\), its range space is always the null-space of \(J\). Therefore, in the projected motion equations (8), any null-space projector can be used. In particular the non-weighted version \(N_J \equiv N_{JI} = I - J^T J\), with pseudoinverse \(J^+ := J^T(JJ^*)^{-1}\), can be applied. Then, with \(N_j = N_J^T\),
\[
N_J(M(q) \ddot{q} + C(q, q) \ddot{q} + P(q) + Q(q, q, t) - u(q, t)) = 0. \tag{9}
\]
3.3 Motion Equations in terms of Minimal Coordinates –Voronets Equations

Due to the \( m \) constraints, only \( n - m \) coordinates are independent (assuming non-redundant constraints). In contrast to the above formulation, the selection of a minimal set of independent generalized coordinates is not possible globally, but only feasible locally at a given configuration \( \mathbf{q} \). Assume that the c-space \( V \) is a smooth manifold at the considered configuration, i.e. \( \delta_{\text{loc}}(\mathbf{q}) = n - m \), and \( n - m \) of the coordinates can be used to parameterize the MBS configuration. Denote with \( \mathbf{s} \in \mathbb{R}^{n-m} \) the vector of (locally) independent minimal coordinates. The coordinate vector can be rearranged as \( \mathbf{q} = (\mathbf{p}, \mathbf{s}) \), where \( \mathbf{p} \) is the vector of \( m \) dependent coordinates, and the Jacobian reads accordingly \( \mathbf{J} = (\mathbf{J}_p, \mathbf{J}_s) \). Then an orthogonal complement for the constraint Jacobian is given explicitly as

\[
\mathbf{F} = \begin{pmatrix}
-\mathbf{J}_p^{-1}\mathbf{J}_s \\
\mathbf{I}_{n-m}
\end{pmatrix}.
\]

That is, \( \dot{\mathbf{q}} = \mathbf{F}\mathbf{s} \) and \( \mathbf{JF} \equiv 0 \). Premultiplication of (3) with \( \mathbf{F}^T \) yields a system of \( n - m \) equations

\[
\mathbf{F}^T(\mathbf{q})(\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\dot{\mathbf{q}}, \mathbf{q})\dot{\mathbf{q}} + \mathbf{P}(\mathbf{q}) + \mathbf{Q}(\mathbf{q}, \mathbf{q}, t) - \mathbf{u}(\mathbf{q}, t)) = 0.
\]

Together with (4) this forms a system of \( n \) independent motions equations. This can be reduced to the minimal coordinates by replacing \( \dot{\mathbf{q}} = \mathbf{F}\mathbf{s} \), which yields the Voronets equations

\[
\overline{\mathbf{M}}(\mathbf{q})\ddot{\mathbf{s}} + \overline{\mathbf{C}}(\dot{\mathbf{q}}, \mathbf{q})\dot{\mathbf{s}} + \overline{\mathbf{P}}(\mathbf{q}) + \overline{\mathbf{Q}}(\mathbf{q}, \mathbf{q}, t) - \overline{\mathbf{u}}(\mathbf{q}, t) = 0
\]

with

\[
\overline{\mathbf{M}} := \mathbf{F}^T\mathbf{M}\mathbf{F}, \quad \overline{\mathbf{C}} := \mathbf{F}^T(\mathbf{C}\mathbf{F} + \mathbf{M}\mathbf{F}), \quad \overline{\mathbf{P}} := \mathbf{F}^T\mathbf{P}, \quad \overline{\mathbf{Q}} := \mathbf{F}^T\mathbf{Q}, \quad \overline{\mathbf{u}} := \mathbf{F}^T\mathbf{u}.
\]

This is a system of \( n - m \) second order ODEs in the \( n - m \) independent coordinates \( \mathbf{s} \). \( \mathbf{M} \) is the mass matrix projected on the subspace of minimal coordinates. The coordinate formulation (12) goes back to Voronets [17], which is a special case of the Maggi equations [3] in holonomic coordinates.

The problematic point with any minimal coordinate formulation is that a chosen set of independent coordinates (determining the orthogonal complement) is only valid within a certain part of the c-space. A set of coordinates fails as local parameters when the matrix \( \mathbf{J}_p \) becomes singular. For the problem addressed in this paper this is not problematic, however.

4 Compatibility Condition for general Switching Constraints

4.1 General Form of Switching Constraints

W.l.o.g. a constraint switching at time \( t = 0 \) is considered, which can be formulated as

\[
h(\mathbf{q}, t) = \begin{cases}
\quad h_-(\mathbf{q}), & t < 0 \\
\quad h_+(\mathbf{q}), & t \geq 0
\end{cases}, \quad \mathbf{J}(\mathbf{q}, t) = \begin{cases}
\quad \mathbf{J}_-(\mathbf{q}), & t < 0 \\
\quad \mathbf{J}_+(\mathbf{q}), & t \geq 0
\end{cases}
\]

where \( h_- \) and \( h_+ \) are respectively the constraints before and after the switching event, and \( \mathbf{J}_- \) and \( \mathbf{J}_+ \) are the corresponding constraint Jacobians. Denote the switch configuration with \( \mathbf{q}_0 := \mathbf{q}(0) \). It is assumed that the individual constraints \( h_- \) and \( h_+ \) are smooth at \( t = 0 \).

4.2 Kinematic Compatibility

The velocity before the switching event is denoted with \( \dot{\mathbf{q}}_- \), and the one after the event with \( \dot{\mathbf{q}}_+ \). The velocity constraints are satisfied at any time, i.e.

\[
\mathbf{J}_+(\mathbf{q}_0)\dot{\mathbf{q}}_+ = 0 \quad (15)
\]

\[
\mathbf{J}_-(\mathbf{q}_0)\dot{\mathbf{q}}_- = 0. \quad (16)
\]

These two conditions can be combined to the following condition on the velocity jump

\[
\mathbf{J}_+(\mathbf{q}_0)\Delta\mathbf{q} + \mathbf{J}_+(\mathbf{q}_0)\dot{\mathbf{q}}_- = 0.
\]

(17)
4.3 Momentum Balance

The switching causes impulsive constrain forces, and thus a discontinuous change of the generalized momentum. This cannot be captured by the projected EOM (8) or (12) since one particular set of constraints is incorporated in order to eliminate the Lagrange multipliers. Starting instead from the unconstrained equations, imposing the quasi-scleronomic constraints leads to the Lagrange equation

\[ M(q) \ddot{q} + C(q,q) \dot{q} + P(q) + Q(q,q,t) + \dot{J}^T(q,t) \lambda = u(q,t). \]  

(18)

The switching causes a jump in the velocity and thus in the generalized momentum due to the constraint Jacobian in (14). It is assumed that \( J_+ \) and \( J_- \) are smooth for \( t \in (-\varepsilon, 0) \) and \( t \in [0, \varepsilon) \), respectively. The momentum conservation at the switching time \( t = 0 \), with corresponding configuration \( q_0 = q(0) \), leads to

\[ \int_0^\varepsilon (M(q) \ddot{q} + C(q,q) \dot{q} + P(q) + Q(q,q,t) + \dot{J}^T(q,t) \lambda) dt = \int_0^\varepsilon u(q,t) dt \]  

(19)

for \( \varepsilon \to 0 \). Noting that \( M \) is constant during the switching, and that \( J_+^T \) is smooth at \( t \in [0, \varepsilon) \) yields the relation

\[ M(q_0) \Delta \dot{q} + \int_0^\varepsilon (C(q,q) \dot{q} + P(q) + Q(q,q,t)) dt + \int_0^\varepsilon J_+^T(q_0,t) \lambda dt = U(q_0) \]  

(20)

on the velocity jump \( \Delta \dot{q} := \dot{q}_+ - \dot{q}_- \). Here \( \Lambda := \int_0^\varepsilon \lambda dt \) is the impulsive constraint force due to the constraint switching, and \( U(q_0) = \int_0^\varepsilon u(q,t) dt \). The integral over \( P \) vanishes for \( \varepsilon \to 0 \) since the potential forces depend smoothly on \( q \). The vector of Coriolis and centrifugal forces \( C(q,q) \dot{q} \) is a quadratic form in \( \dot{q} \). The \( i^{th} \) component of this force reads explicitly \( \sum_{j,k} \Gamma_{ijk} (q) \dot{q}^i \dot{q}^j \dot{q}^k \), with \( \Gamma_{ijk} \) being the Christoffel symbols corresponding to the mass matrix. Although the velocities are discontinuous during the event, they are bounded. Hence also the integral over the quadratic form \( C(q,q) \dot{q} \) vanishes for \( \varepsilon \to 0 \). The only remaining integral term is that for \( Q \) representing general velocity dependent forces, such as friction.

**Assumption 1.** *The contribution of the forces \( Q \) to the momentum balance can be neglected.*

With this presumption the momentum balance becomes

\[ M(q_0) \Delta \dot{q} + J_+^T(q_0) \Lambda = U(q_0). \]  

(21)

4.4 Overall Transition Condition

Combining the momentum balance with the kinematic conditions (17) gives rise to the following system

\[ \begin{pmatrix} M(q_0) & J_+^T(q_0) \\ J_+(q_0) & 0 \end{pmatrix} \begin{pmatrix} \Delta \dot{q} \\ \Lambda \end{pmatrix} = \begin{pmatrix} U(q_0) \\ -J_+(q_0) q_- \end{pmatrix}. \]  

(22)

A solution \( \Delta \dot{q} \) of (22) is an admissible change of generalized velocity such that the constraints prior and after the switching event as well as the momentum balance are satisfied. This system of \( n + m_+ \) equations for the \( n + m_+ \) unknowns \( \dot{q}_+ \) and \( \Lambda \), has a unique solution provided that the constraints are non-redundant.

4.5 Incorporation into a Time Integration Scheme

The transition condition (22) is only required at a switching event. Any motion equations (such as (3), (7) or (12)) can be numerically integrated until the event, providing the velocity \( \dot{q}_- \) prior to the event and the configuration \( q_0 \in V \) satisfying the geometric constraints. The velocity jump at \( t = 0 \) could then be determined by solving (22). The compatibility condition (22) yields the full (redundant) state of the system, from which any desired set of coordinates can be selected, e.g. minimal coordinates.
The coefficient matrix in (22) is exactly the one in the index 1 formulation (5) evaluated at \( \mathbf{q}_0 \). Hence, the system (22) can be solved in parallel with the EOM if the EOM (5) is integrated. However, for large \( n \) and a large number \( m \) of constraints (e.g. switching molecular systems [13]) the EOM in redundant or minimal coordinates may be computationally simpler. Then the condition (22) is still applicable, but would require solving a system of \( n + m \) equations. To avoid this, a transition condition for either formulation is required. This can be derived for the special case of activation of additional constraint, as shown in the following.

The impulsive force \( \mathbf{U}(\mathbf{q}_0) \) can possibly be substituted by a contact model.

5 Compatibility Condition for Activation of additional Constraints

5.1 Activation of additional Constraints

A frequently encountered type of VTM switching constraints is that further constraints are activated in addition to constraints already active before the switching event. This can be modeled by the following form of the quasi-scleronomic constraints

\[
h(q,t) = \begin{cases} h^-(q) := h_1(q), & t < 0 \\ h^+(q) := \begin{pmatrix} h_1(q) \\ h_2(q) \end{pmatrix}, & t \geq 0 \end{cases}, \quad \mathbf{J}(q,t) = \begin{cases} J^-(q) := J_1(q), & t < 0 \\ J^+(q) := \begin{pmatrix} J_1(q) \\ J_2(q) \end{pmatrix}, & t \geq 0 \end{cases}.
\]

(23)

Here \( h_1(q) = 0 \) is a set of \( m_1 \) persistent constraints throughout the switching to which another set of \( m_2 \) constraints \( h_2(q) = 0 \) is added at \( t = 0 \).

The c-space before the switching event is \( V_- := h_1^{-1}(0) \) and that after is \( V_+ := h_2^{-1}(0) \). Due to the persistent constraints it is \( V_- := h_1^{-1}(0) = h_1^{-1}(0) \) and \( V_+ := h_2^{-1}(0) = h_1^{-1}(0) \cap h_2^{-1}(0) \), and thus \( V_+ \subset V_- := h_1^{-1}(0) \).

Denote with \( \mathbf{q}_0 := \mathbf{q}(0) \) the configuration where the switching takes place at \( t = 0 \).

Assumption 2. Both constraint mappings, \( h_1 \) and \( h_2 \), are smooth at \( \mathbf{q}_0 \). That is, \( V_1 = h_1^{-1}(0) \) and \( V_2 = h_2^{-1}(0) \) are locally smooth manifolds at \( \mathbf{q}_0 \). Further \( J_1(\mathbf{q}_0) \) and \( J_2(\mathbf{q}_0) \) have full rank \( m_1 \) and \( m_2 \), respectively. That is, the constraints are non-redundant.

Assumption 3. The constraints \( h_1 \) and \( h_2 \) are locally algebraically independent. That is, \( V_+ = V_1 \cap V_2 \), with \( V_1 = h_1^{-1}(0) \) and \( V_2 = h_2^{-1}(0) \), is a smooth manifold of dimension \( n - m_1 - m_2 \).

5.2 Transition Condition for the Lagrangian Motion Equations

With the partitioned form of the constraints (23), the transition conditions (5) are

\[
\begin{pmatrix} \mathbf{M}(\mathbf{q}_0) & \mathbf{J}_1^T(\mathbf{q}_0) & \mathbf{J}_2^T(\mathbf{q}_0) \\ \mathbf{J}_1(\mathbf{q}_0) & 0 & 0 \\ \mathbf{J}_2(\mathbf{q}_0) & 0 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{\Delta q} \\ \Lambda_1 \\ \Lambda_2 \end{pmatrix} = \begin{pmatrix} \mathbf{U}(\mathbf{q}_0) \\ -\mathbf{J}_1(\mathbf{q}_0) \mathbf{\Delta q}_- \\ -\mathbf{J}_2(\mathbf{q}_0) \mathbf{\Delta q}_- \end{pmatrix}
\]

(24)

where \( \Lambda_1 \) and \( \Lambda_2 \) are the impulsive constraint forces due to the persistent and additional constraints, respectively.

- Solving the system (24) requires inverting the \( (n + m_1 + m_2) \times (n + m_1 + m_2) \) matrix. The mass matrix \( \mathbf{M} \) in (24) can be inverted with the same method as for the solution of the system (5) within the time integration scheme. For instance, if the EOM are evaluated with an \( O(n) \) method, then also the system (24) can be evaluated with this method.

- The condition can be used in conjunction with any dynamics formulation (redundant or minimal coordinates). The compatibility condition (32) yields the full (redundant) state of the system, from which any desired set of coordinates can be selected, e.g. minimal coordinates. The system (24) is only solved at the switching event.

- If the constraints \( h_1(q) = 0 \) and \( h_2(q) = 0 \) are independent (assumption 3), and \( \mathbf{J}_1 \) and \( \mathbf{J}_2 \) have full rank (assumption 2), the coefficient matrix in (24) has full rank \( n + m_1 + m_2 \).
5.3 Transition Condition in Terms of Redundant Coordinates

5.3.1 Momentum Balance

In the projected EOM (8), the Lagrange multipliers are eliminated so that the above approach is not applicable directly. Nevertheless the constraint forces can be retained if the projection in the equations (8) is restricted to the set of persistent constraints \( h_1(q) = 0 \). Imposing the additional constraints leads to corresponding constraint forces. The projected EOM accounting for the persistent constraints are

\[
N_{J_{r},M}^T(M(q) \ddot{q} + C(q,q)q + P(q) + Q(q,q,t) + J^T(q,t)\lambda - u(q,t)) = 0
\]  

(25)

that split to the two equations

\[
N_{J_{r},M}^T(M(q) \ddot{q} + C(q,q)q + P(q) + Q(q,q,t) - u(q,t)) = 0, \quad \text{for } t < 0
\]

\[
N_{J_{r},M}^T(M(q) \ddot{q} + C(q,q)q + P(q) + Q(q,q,t) + J^T(q)\lambda_2 - u(q,t)) = 0, \quad \text{for } t \geq 0.
\]

With \( Jq = -J\dot{q} \) the equations (25) can be written as

\[
M(q)\ddot{q} + N_{J_{r},M}^T(q)\left(C(q,q)q + P(q) + Q(q,q,t) + J^T(q,t)\lambda\right) + J^T_1(J_1M^{-1}J^T_1)^{-1}J_1\dot{q} = N_{J_{r},M}^T(q)\ddot{q}.
\]

(26)

The momentum balance then reads

\[
\int_0^\varepsilon M(q)\ddot{q}dt + \int_0^\varepsilon N_{J_{r},M}^T(q)\left(C(q,q)q + P(q) + Q(q,q,t)\right)dt + \int_0^\varepsilon N_{J_{r},M}^T(q)J^T_2(q)\lambda_2 dt = U(q_0).
\]

(27)

With the assumed smoothness of the persistent constraints (assumption 2), \( N_{J_{r},M} \) is smooth, and neglecting the integral of \( Q \) (assumption 1), the second integral term vanishes. Presuming smoothness of the activated constraints for \( t \geq 0 \), then (27) yields

\[
M(q_0)\Delta\dot{q} + N_{J_{r},M}^T(q_0)J^T_2(q_0)\lambda_2 = U(q_0)
\]

(28)

with the impulsive constraint force \( \lambda_2 := \int_0^\varepsilon \lambda_2 dt \).

If \( u \) is smooth at \( t_0 \) (no discontinuous control forces), the impulsive force \( U(q_0) \) vanishes.

5.3.2 Kinematic Compatibility

The persistent constraints can be written as

\[
J_1(q_0)\dot{q} = J_1(q_0)\dot{q}_+ - J_1(q_0)\Delta\dot{q} = 0,
\]

(29)

and the additional constraints for \( t \geq 0 \) as

\[
J_2(q_0)\dot{q}_+ = J_2(q_0)\Delta\dot{q} + J_2(q_0)\dot{q}_- = 0.
\]

(30)

The constraints (29) must be satisfied throughout the switching. A particular solution of (29) is \( \dot{q}_+ = J^T_1(q_0)J_1(q_0)\Delta\dot{q} \). Inserting this into (30) yields

\[
J_2(q_0)N_{J_{r},M}(q_0)\Delta\dot{q} = -J_2(q_0)\dot{q}_-.
\]

(31)

5.3.3 Overall Transition Condition

The momentum balance together with the kinematic compatibility conditions can be summarized as

\[
\begin{pmatrix}
M(q_0) \\
J_2(q_0)N_{J_{r},M}(q_0)
\end{pmatrix}
\begin{pmatrix}
\Delta\dot{q} \\
\Lambda_2
\end{pmatrix}
= \begin{pmatrix}
U(q_0) \\
-J_2(q_0)\dot{q}_-
\end{pmatrix}.
\]

(32)

The simple case of an MBS subjected to a single set of constraints, which is activated at \( t = 0 \), is included with \( N_{J_{r},M} = I \). The so determined \( \Delta\dot{q} \) is the admissible velocity jump such that the momentum balance is satisfied. The impulsive constraint force \( \Lambda_2 \) represents the change in the system’s generalized momentum. As a by-product, the impulsive constraint force \( \Lambda_2 \) caused by the event is determined.
• The compatibility condition (32) yields the full (redundant) state of the system. It can be used in conjunction with any dynamics formulation.

• The square coefficient matrix has dimension $n + m_2$ since only the $m_2$ (projected) constraints are included that are added at the even. Solving (32) requires inversion of the $(n + m_2) \times (n + m_2)$ matrix. The determination of the null-space projector requires inversion of a $m_1 \times m_1$ matrix. This inverse exists with assumption 2.

• The system (32) is solved at the switching event. If the EOM are evaluated with an $O(n)$ method, then possibly also the system (32) can be solved using this method. The apparent obstacle is the null-space projector. Evaluating the null-space projector requires computing the pseudoinverse $J_M^+ = M^{-1} J^T (J M^{-1} J^T)^{-1}$, and thus the inverse of the mass matrix. The complexity depends on the specific form of the constraints. These are very simple for instance, if joints are locked (see section 6).

• If the constraints $h_1(q) = 0$ and $h_2(q) = 0$ are independent, i.e. ker$J_1 \cap$range $J_2 = 0$ (assumption 3), and $J_2$ has full rank (assumption 2), then the coefficient matrix in (32) has full rank $n - m_2$.

5.4 Transition Condition in Terms of Minimal Coordinates

5.4.1 Momentum Balance

The above condition (32) involves the null-space projector, and thus requires the determination of $M^{-1}$. Even if this is only necessary at the switching points, for large systems this may be infeasible to be performed. An alternative formulation can be derived in terms of minimal coordinates resorting to the concept of Voronets equations (12). The latter cannot be applied unaltered to quasi-scleronomic constraints, however. For the special case of activation of additional constraints, the persistent constraints $h_1(q) = 0$ can be used to formulate EOM on $V_-$. The additional constraints $h_2(q) = 0$ then restrict the EOM to $V_+ \subset V_-$. Starting from the Lagrange equations (18), the Voronets equations on $V_-$ are

$$\bar{M}(q)s + \bar{C}(q,q)s + \bar{Q}(q,q,t) + F_1^T(q)J^T(q,t)\lambda = \bar{u}(q,t)$$

that split into

$$M(q)s + C(q,q)s + Q(q,q,t) = u(q,t), \quad \text{for } t < 0$$

$$\bar{M}(q)s + \bar{C}(q,q)s + \bar{Q}(q,q,t) + F_1^T(q)J_2^T(q)\lambda_2 = u(q,t), \quad \text{for } t \geq 0$$

with $F_1$ being the orthogonal complement to $J_1$, and $\lambda_2$ are the Lagrange multipliers accounting for the additional constraints not captured by the projection to $V_-$. This resembles the formulation in redundant coordinates (25), but now the EOM already projected to $V_-$ are subjected to additionally constraints, whereas in (25) the projector changes. The equations (33) may be considered as constrained Voronets equations. The crucial difference is that here locally valid set of independent coordinates $s$ must be selected explicitly.

Proceeding as above, the momentum balance during the switching leads to

$$\int_0^\varepsilon \bar{M}(q)\dot{s}dt + \int_0^\varepsilon (C(q,q)\dot{q} + Q(q,q,t))dt + \int_0^\varepsilon F_1^T(q)J_2^T(q)\lambda_2dt = U(q_0).$$

It is presumed that $s$ are valid parameters, so that $F$ is smooth, that $J_1$ is full rank so that $Cq$ and $Q$ are smooth. The momentum balance then yields

$$\bar{M}(q_0)\Delta s + F_1^T(q_0)J_2^T(q_0)\lambda_2dt = U(q_0)$$

where $\Delta s = s_+ - s_-$ is the change in the independent generalized velocities, and $\lambda_2 := \int_0^\varepsilon \lambda_2dt$ is the impulsive constraint force collocated with the minimal coordinates.
5.4.2 Kinematic Compatibility

The persistent velocity constraints are already incorporated in the Voronets equations on \( V_\cdot \). It remains to account for the additionally activated constraints. After the switching event these are

\[
0 = J_2(q_0)^T \dot{q}_\cdot = J_2(q_0)^T F_1(q_0) \dot{s}_- \\
= J_2(q_0)^T F_1(q_0) \Delta s + J_2(q_0)^T F_1(q_0) \dot{s}_-.
\]

The last equation relates the independent velocities before the event, \( \dot{s}_- \), to the change due to the switching, \( \Delta s \).

5.4.3 Overall Transition Condition

Combining momentum balance (35) with the kinematic constraints (36) gives rise to the overall condition

\[
\begin{pmatrix}
\mathbf{M}(q_0) & F_1^T(q_0) J_2^T(q_0) \\
J_2(q_0) F_1(q_0) & 0
\end{pmatrix}
\begin{pmatrix}
\Delta s \\
\Lambda_2
\end{pmatrix}
= \begin{pmatrix}
U(q_0) \\
-J_2(q_0) F_1(q_0) \dot{s}_-
\end{pmatrix}.
\]

This is the compatibility condition on the generalized momentum and the generalized velocity. A solution \( \Delta s \) of (37) satisfies the kinematic constraints while ensuring momentum balance. This is a system of \( n - m_1 + m_2 \) equations for the change of \( n - m_1 \) generalized velocities \( \Delta s \) and the \( m_2 \) impulsive constraint forces \( \Lambda_2 \). The system (37) has an obvious interpretation as momentum balance of a constrained MBS with local DOF \( n - m_1 \) that is subjected to \( m_2 \) additional constraints.

- Solving (37) requires inverting the \( (n - m_1 + m_2) \times (n - m_1 + m_2) \) coefficient matrix.
- The computation of the orthogonal complement \( F_1 \) requires inversion of a \( m_1 \times m_1 \) matrix.

6 Example: Joint Locking in a Planar 3-Bar Pendulum

The first example is a planar 3R pendulum with parallel revolute joint axes linked to the ground and subjected to gravity in fig. 1. The solid aluminium links have lengths 1 m and rectangular cross section of 0.2 \( \times \) 0.2 m. The generalized coordinates are the three joint angles. The system is released from rest in the configuration \( q = (\pi/6, \pi/6, \pi/6) \) and set in motion by the gravity force.

\[ q^1 = q^2 = q^3 = \pi/6. \]

Also shown are the configurations at the switching time \( t_1 = 0.8 \) s where joint 2 is locked, and at the switching time \( t_2 = 1.3 \) s where joint 3 is locked.

**Figure 1.** Planar 3R-pendulum in its reference configurations \( q^1 = q^2 = q^3 = \pi/6 \). Also shown are the configurations at the switching time \( t_1 = 0.8 \) s where joint 2 is locked, and at the switching time \( t_2 = 1.3 \) s where joint 3 is locked.
At $t_1 = 0.8\, s$, the second joint is locked. The third joint is locked at $t_2 = 1.4\, s$. This successive joint locking is described by the quasi-scleronomic constraints

$$
 h(q, t) = \begin{cases} 
 0, & t < t_1 \\
 h_1(q), & t_1 \leq t < t_2 \\
 \left( \begin{array}{c}
 h_1(q) \\
 h_2(q)
 \end{array} \right), & t_2 \leq t
\end{cases}
$$

with $h_1(q) = q^2 - q^2(t_1)$, and $h_2(q) = q^3 - q^3(t_2)$, and the corresponding Jacobians

$$
 J_1 = \left( \begin{array}{cc}
 0 & 1 \\
 0 & 0
 \end{array} \right), \quad J_2 = \left( \begin{array}{cc}
 0 & 0 \\
 0 & 1
 \end{array} \right).
$$

(38)

**Minimal Coordinates** Prior to $t_1$, the system is unconstrained, so that the transition condition (22) applies with $J_+ = J_1$. Additionally, joint 3 is locked at $t_2$. Now the compatibility condition (37) is invoked with $J_2$ in (38), and with the orthogonal complement $F_1 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$ to $J_1$. The minimal coordinates are $s = (q^1, q^3)$. The latter are the remaining independent joint variables after the locking of joint 2 at $t_1$. Notice that, when joints in an open kinematic chain are locked, the set of minimal coordinates in the transition equations is unique.

**Redundant Coordinates** When (32) is used as transition condition for the additional joint locking at $t_2$, the null-space projector to $J_1$, and thus the pseudoinverse is required. From the special form of $J_1$, follows

$$
 J_1^\perp = M^{-1} J_1^T \left( J_1 M^{-1} J_1^T \right)^{-1} = \begin{pmatrix} \bar{m}_{12}/\bar{m}_{22}, 1, \bar{m}_{32}/\bar{m}_{22} \end{pmatrix}^T,
$$

where $\bar{m}_{ij}$ are the entries of the inverse $M^{-1}$, and thus

$$
 N_{J_1,M} = \begin{pmatrix} 1 & -\bar{m}_{12}/\bar{m}_{22} & 0 \\
 0 & 0 & 0 \\
 0 & -\bar{m}_{32}/\bar{m}_{22} & 1
 \end{pmatrix}
$$

**Results** The motion equations are integrated, and the transition conditions are invoked at the switching events at $t_1$ and $t_2$. Both formulations (32) and (37) are evaluated, and lead to identical results. The time evolution of the generalized coordinates is shown in fig. 2a), and the corresponding generalized velocities in fig. 2b). The velocity jumps are such that the momenta of the unlocked joints is continuous. Fig. 3a) shows the components of the generalized momentum vector $P = M(q) \dot{q}$. At the switching points, the total energy jumps whereas it is preserved otherwise (fig. 3b)). Notice that the geometric locking constraint is exactly satisfied because the transition conditions (32) yields $\dot{q}^2 = 0$ which is exactly integrated to $q^2 = q^2(t_1)$, for $t \geq t_1$, and analogously for $q^3$.

![Figure 2. a) Joint angles, and b) joint rates of the planar 3R-pendulum.](image-url)
7 Conclusion

The time integration of the motion equations governing the dynamics of variable topology mechanisms (VTM) must ensure the compatibility of the generalized momentum and velocity at the switching events. In this paper a compatibility condition for the general case of successive activation of multiple constraints is presented. The condition is formulated in terms of the full set of (due to the constraint activation) redundant coordinates as well as in terms of minimal coordinates. The formulations have different computational complexity. Both formulations can be evaluated independently from the actual time integration method. The method can be easily extended to account for non-holonomic velocities $v$ related to generalized speed by $v = A(q) \dot{q}$.

As example, the locking of joints of a 3R mechanism is reported. The method has been applied to various other systems, including a metamorphic robotic hand whose palm consists of a spherical 5-bar linkage imitating the inherent mobility of the human hand. Other possible applications include docking maneuvers.

The formulation applies also to MBS comprising flexible bodies, in which case the generalized coordinates include modal/nodal coordinates. Then, as always when constraining flexible MBS, the constraints may lead to artificial mode locking.

Aknowledgement

The author acknowledges that this work has been partially supported by the Austrian COMET-K2 program of the Linz Center of Mechatronics (LCM).

REFERENCES


1 ABSTRACT
This paper presents the mathematical framework for the efficient construction of the equations of motion of complex multibody problems when uncertainty exists in the systems’ parameters and/or inputs. Herein, uncertainty is prorogated through the system dynamics by using the method of polynomial chaos expansion (PCE). In this scheme, states of the system are projected onto the space of appropriate orthogonal base functions. Furthermore, the method of Divide-and-Conquer Algorithm (DCA) is extended to construct the equations of motion of the resulting non-deterministic system. In this scheme, the mathematical formulation to generate the projected handle equations of motion of the bodies and the projected constraint equations of the connecting joints are constructed in terms of the PCE. Finally, these projected equations are used to perform the assembly and disassembly passes to form and solve the equations of motion. The proposed method is highly parallelizable and scales down the computational complexity as a linear and logarithmic function of the state variables in serial and parallel implementations, respectively.

Keywords: Divide-and-Conquer Algorithm, Polynomial Chaos Expansion, Multibody Dynamics, Uncertainty Quantification.

2 INTRODUCTION
Although deterministic approaches are widely used to model electromechanical systems, such problems perform under some degree of uncertainty. Uncertainty may originate from the lack of the knowledge in the physical and/or geometric properties of the system such as the body dimensions and mass, and stiffness and damping coefficients. Further, inputs to the system to control its dynamic behavior may suffer from some degree of uncertainty. As such, there is a critical need to develop advanced methods to efficiently and accurately capture the dynamics of the system under the influence of uncertainty.

In the field of dynamic analysis of multibody problems, the method of polynomial chaos expansion was applied to study uncertainty in multibody systems [1, 2]. In another effort, uncertain rigid bodies for rotor dynamical systems were studied by using the nonparametric probabilistic approach [3, 4] consisting in replacing the mass and gyroscopic matrices by random matrices [5, 6]. The mass, the center of mass and the tensor of inertia of the rigid bodies were replaced by random variables in [7]. Then prior probability distributions of the stochastic model have been constructed by using the maximum entropy principal under the constraints defined by the available information. Finally, Baysian networks have been used for the state estimation of mechanisms [8].

The polynomial chaos expansion method can appropriately capture the dynamic behavior of non-deterministic systems containing uncertainties with large amplitudes. This technique has been successfully applied to different fields including, stochastic finite elements, thermo-fluid applications, flow-structure interactions, nonlinear estimation, probabilistic robust control, systems biology, control, and solid mechanics [9]-[15]. As the size of the problem increases in this approach, using this technique along with the traditional methods of forming the equations of motion of constrained multibody systems in the differential algebraic equations (DAEs) framework prevents
Table 1: Relations between base functions, supports, and distribution functions [16]

<table>
<thead>
<tr>
<th>Distribution</th>
<th>Density function</th>
<th>Base function</th>
<th>Support range</th>
<th>Weight function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uniform</td>
<td>( \frac{1}{2} )</td>
<td>Legendre</td>
<td>([-1, 1])</td>
<td>1</td>
</tr>
<tr>
<td>Normal</td>
<td>( \frac{1}{\sqrt{2\pi}}e^{-\frac{x^2}{2}} )</td>
<td>Hermite</td>
<td>((-\infty,\infty))</td>
<td>(e^{-\frac{x^2}{2}})</td>
</tr>
</tbody>
</table>

performing the simulation in a timely effective manner. This prohibits the efficient reliability assessment of heavily constrained multibody problems.

The PCE technique has been used in [16, 17] in conjunction with the divide-and-conquer algorithm to reduce the computational complexity of forming and solving the equations of motion of the heavily constrained non-deterministic multibody problems. In that method, all random variables are symbolically carried through the entire process of forming and solving the governing equations. This paper extends and improves the PCE-based DCA to remove this shortcoming. In this scheme, the mathematical framework to project the handle equations of motion of the system onto the space of appropriate base functions and the mechanism to generate the projected governing constraint equations is provided. Unlike the previous technique, these equations do not contain symbolic random variables. Finally, the mathematical framework to perform the assembly and disassembly processes to form and solve the projected equations of motion of the system are presented.

3 UNCERTAINTY ANALYSIS IN THE POLYNOMIAL CHAOS EXPANSION SCHEME

Using the PCE method, all stochastic random inputs and response outputs are projected onto the space of independent orthogonal polynomial base functions selected according to the nature of the distribution of random parameters of the system as shown in Table 1. For instance, consider a second order random process \( C(\xi,t) \) with finite variance. This quantity is then expressed in terms of \( \tilde{C}_i \) as time dependent modal coefficients, and \( \Psi_i(\xi_1, \xi_2, ..., \xi_n) \) as time-invariant generalized polynomial chaos of order of \( n(i) \)

\[
C(\xi,t) = \sum_{i=0}^{\infty} \tilde{C}_i(t)\Psi_i(\xi) \approx \sum_{i=0}^{N_t} \tilde{C}_i(t)\Psi_i(\xi). \tag{1}
\]

Herein, \( \Psi_i(\xi_1, \xi_2, ..., \xi_n) \) are functions of multidimensional random variables \( \xi(\xi_1, \xi_2, ..., \xi_n) \). These polynomials form a complete orthogonal basis for the Hilbert space of square integrable random variables based of the weight functions provided in Table 1. As such, the weighted inner product using the joint probability density of the random variable \( w(\xi) \) associated with each specific distribution shown in Table 1 is determined by

\[
<\Psi_i(\xi),\Psi_j(\xi)> = \int \Psi_i(\xi)\Psi_j(\xi)w(\xi)d\xi = h^2\delta_{ij}, \tag{2}
\]

where \( h^2 \) is a positive number and \( \delta_{ij} \) is the Kronecker delta which is 1 for \( i = j \), and zero for \( i \neq j \). Furthermore, for a function \( f \), one can define the Galerkin projection as

\[
<f,\Psi_j(\xi)> = \int f\Psi_j(\xi)w(\xi)d\xi. \tag{3}
\]

Because the expression in Eq. (1) converges to any random process in \( L^2 \) sense [18], in practice, one truncates this infinite expansion at a finite number of random variables and a finite expansion order. Using an expansion of total order \( P \), the total number of modal terms in the summation for a system with \( N_u \) finite number of uncertain variables \( \xi = (\xi_1, \xi_2, ..., \xi_{N_u}) \), [1, 15], i.e. \( N_t \), becomes [15]

\[
N_t + 1 = \frac{(N_u + P)!}{N_u!P!}. \tag{4}
\]
Finally, stochastic parameters of the system can be extracted from modal coefficients. For instance, the expected value and variance of $C$ are calculated as

$$ E[C] = \bar{C}_0, $$

$$ \text{Var}(C) = E[C^2] - (E[C])^2 = \sum_{i=1}^{\infty} \bar{C}_i^2 < \Psi_i^2 >. $$

### 4 TRADITIONAL PCE-BASED DIFFERENTIAL ALGEBRAIC EQUATIONS (DAEs)

In general, the equations governing dynamics of a constrained multibody system may be expressed as follows:

$$ u = C(q,t)\dot{q} + D(q), $$

$$ M(q,t;s)\ddot{q} = \Phi_{,q}^T(q,t,s)\lambda + K(q,\dot{q},t;s), $$

$$ \Phi(q,t;s) = 0. $$

In this set of equations, $q$ is a column matrix of the generalized coordinates of the system. Similarly, $t$ represents time, and $s$ is the set of all system design parameters. Equation (7) represents the mapping between the generalized speeds of the system $u$ and the first time derivative of the generalized coordinates $\dot{q}$. The term $M$ in Eq. (8) is the mass matrix of the entire system while $\Phi_{,q}$ represents the constraint Jacobian associated with the partial derivative of the $m$ independent algebraic constraint equations. Finally, $K$ is the column vector of applied and body loads, as well as centripetal and Coriolis terms [17].

Consider a multibody system containing uncertain parameters ${s_k}_{k=1}^{k=N_u}$. Uncertainty in system’s parameters affects the system’s states including generalized coordinates, generalized speeds, and time derivative of generalized speeds. Assuming the generalized speeds of the system as the time derivative of the system’s generalized coordinates, one can express the stochastic equations of motion of a constrained multibody system as [16]

$$ u_{ij}(t) = \dot{q}_{ij}(t), \quad (i = 1, \ldots, n), \quad (j = 0, \ldots, N_t) $$

$$ \sum_{j=0}^{N_t} \sum_{m=0}^{N_c} M_{ij}(\sum_{m=0}^{N_c} q_m(t)\Psi_m(\xi), \sum_{m=0}^{N_c} s_m\Psi_m(\xi)) \cdot \dot{u}_j(t)\Psi_j(\xi) = $$

$$ K(t, \sum_{m=0}^{N_c} q_m(t)\Psi_m(\xi), \sum_{m=0}^{N_c} u_m(t)\Psi_m(\xi), \sum_{m=0}^{N_c} s_m\Psi_m(\xi)) + $$

$$ \Phi_{,q}(\sum_{m=0}^{N_c} q_m(t)\Psi_m(\xi), \sum_{m=0}^{N_c} s_m\Psi_m(\xi)) = 0, $$

where $s_m = [s_{1m}, \ldots, s_{N_m}]^T$, $q_m = [q_{1m}, \ldots, q_{Nm}]^T$, $u_m = [u_{1m}, \ldots, u_{Nm}]^T$, $\dot{u}_m = [\dot{u}_{1m}, \ldots, \dot{u}_{Nm}]^T$, $\lambda_m = [\lambda_{1m}, \ldots, \lambda_{Nm}]^T$, $m = 0, \ldots, N_t$.

To study the effect of the uncertainty on the system dynamics, one may use the Galerkin projection [1] to project Eqns. (11) and (12) onto the space spanned by the base functions $[\Psi_0, \ldots, \Psi_{N_t}]$. The resulting equations are then expressed in the matrix form as

$$
\begin{bmatrix}
\mathcal{M}_{n(N_t+1)\times n(N_t+1)} & \mathcal{B}_{n(N_t+1)\times n(N_t+1)}^T \\
\mathcal{B}_{n(N_t+1)\times n(N_t+1)} & 0_{n(N_t+1)\times n(N_t+1)}
\end{bmatrix}
\begin{bmatrix}
\dot{u}_{10}, \ldots, \dot{u}_{nN_t}, \lambda_{10}, \ldots, \lambda_{nN_t}
\end{bmatrix}^T
= \begin{bmatrix}
\mathcal{X}_1 \\
\mathcal{X}_2
\end{bmatrix}.
$$
Table 2: Number of unknown modal terms in the polynomial chaos expansion $N_t + 1$

<table>
<thead>
<tr>
<th>$N_u$</th>
<th>$P=1$</th>
<th>$P=2$</th>
<th>$P=3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>6</td>
<td>21</td>
<td>56</td>
</tr>
<tr>
<td>10</td>
<td>11</td>
<td>66</td>
<td>286</td>
</tr>
<tr>
<td>20</td>
<td>21</td>
<td>231</td>
<td>1771</td>
</tr>
</tbody>
</table>

The overall computational cost associated with solving Eq. (13) for such systems at each evaluation is $(n(N_t + 1))^3 + (n(N_t + 1))^2(n_c(N_t + 1)) + (n(N_t + 1))(n_c(N_t + 1))^2 + (n_c(N_t + 1))^3$. According to the values of $N_t + 1$ shown in Table 2, using traditional methods of forming and solving the stochastic DAEs involving a large number of generalized coordinates, and independent constraints equations computationally becomes intractable. For instance, for a small non-constrained system with 20 generalized coordinates and 20 uncertain parameters, if one uses polynomial chaos up to the order 2, the number of the states of the stochastic system rapidly increases to $20 \times 231$. This undesirable increase in the simulation cost will prevent the uncertainty analysis as the size of the system increases [16].

5 PCE-BASED DIVIDE AND CONQUER ALGORITHM

Since the system size increases as one uses the PCE to model the uncertainty, this paper extends the method of Divide and Conquer Algorithm (DCA) to form the PCE-based equations of motion. The DCA is implemented using a series of recursive assembly and disassembly processes to form and then solve the equations of motion of the system. The interested reader is referred to [19],[20]-[29] for more detail on the development and application of the DCA in modeling and simulation of complex multi-rigid/flexible-body systems.

5.1 Preliminary Mathematics

Since in the DCA, one works with spatial quantities including velocities, accelerations, and forces, we need to come up with a convenient expression of such quantities in the PCE-framework. For instance, consider a spatial quantity $R \in \mathbb{R}^6$ ($R$ is the $6 \times 1$ column matrix). This quantity is expressed as

$$R(\xi, t) \approx \sum_{i=0}^{N_t} \tilde{R}_i(t)\Psi_i(\xi),$$  \hspace{1cm} (14)

where $\tilde{R}_i \in \mathbb{R}^6$ is also expressed as

$$\tilde{R}_i = [\tilde{R}_i^1, ..., \tilde{R}_i^6]^T.$$  \hspace{1cm} (15)

Performing the Galerkin projection on Eqn. (14), one can obtain the column matrix $\tilde{R}_j \in \mathbb{R}^6$ as

$$\tilde{R}_j = \frac{<R_{6 \times 1}, \Psi_j>}{h_j^2}, \quad j = 0, 1, ..., N_t.$$  \hspace{1cm} (16)

This relation is now implemented for all of the values of $j$ ($j = 0, ..., N_t$) to find the extended column matrix $\tilde{R} \in \mathbb{R}^{6(N_t+1)}$ as

$$\tilde{R}_{6(N_t+1) \times 1} = [\tilde{R}_0, ..., \tilde{R}_j, ..., \tilde{R}_{N_t}]^T = \left[\frac{<R_{6 \times 1}, \Psi_0>}{h_0^2}, ..., \frac{<R_{6 \times 1}, \Psi_j>}{h_j^2}, ..., \frac{<R_{6 \times 1}, \Psi_{N_t}>}{h_{N_t}^2}\right]^T.$$  \hspace{1cm} (17)

5.2 Projected Handle Equations of Motion of a Single Body

A handle on body $k$ shown in Fig. 1 is a point of the body, such as the joint location $H_1^k$ and $H_2^k$, through which it has an interaction with the environment. For a deterministic system, the
two-handle equations of motion of body $k$ are expressed as [24]

$$A_1^k = \eta_{11}^k F_{1c}^k + \eta_{12}^k F_{2c}^k + \eta_{13}^k$$

$$A_2^k = \eta_{21}^k F_{1c}^k + \eta_{22}^k F_{2c}^k + \eta_{23}^k$$

where spatial accelerations of the handles, $A_1^k$ and $A_2^k$, are expressed in terms of unknown spatial constraint forces $F_{1c}^k$ and $F_{2c}^k$, applied to these handles of the body due to the kinematic constraint at the connecting joint. The terms $\eta_{ij}^k (i,j = 1,2)$ are associated with the inertia of the body, while $\eta_{ij}^k (i = 1,2)$ represent the column matrices corresponding to the known spatial applied forces, as well as centripetal and Coriolis terms.

In the presence of uncertainty, all of the ordination angles, generalized coordinates, and their time derivatives are expressed in terms of independent orthogonal base functions. As such, the coefficients $\eta_{ij}^k$ are functions of the uncertain variable $\xi$, and replaced by $\hat{\eta}_{ij}^k (i,j = 1,2)$ and $\hat{\eta}_{ij}^k (i = 1,2)$. Furthermore, following Eqn. (14), one can project spatial accelerations of the handles, and corresponding spatial constraint loads onto the space of orthogonal base functions. As such, using modal terms $\hat{A}_1^k, \hat{A}_2^k, \hat{F}_1^k, \hat{F}_2^k$, and $\hat{F}_{2c}^k \in \mathbb{R}^6$, the nondeterministic handle equations of body $k$ are expressed as

$$\sum_{i=0}^{N_1} \hat{A}_{1,i}^k \Psi_i = \hat{\eta}_{11}^k \sum_{i=0}^{N_1} F_{1c,i}^k \Psi_i + \hat{\eta}_{12}^k \sum_{i=0}^{N_1} F_{2c,i}^k \Psi_i + \hat{\eta}_{13}^k$$

$$\sum_{i=0}^{N_1} \hat{A}_{2,i}^k \Psi_i = \hat{\eta}_{21}^k \sum_{i=0}^{N_1} F_{1c,i}^k \Psi_i + \hat{\eta}_{22}^k \sum_{i=0}^{N_1} F_{2c,i}^k \Psi_i + \hat{\eta}_{23}^k$$

Applying the Galerkin projection (see Eq. (3)) to the above equations yields

$$\sum_{i=0}^{N_1} \hat{A}_{1,i}^k < \Psi_i, \Psi_j > = \sum_{i=0}^{N_1} < \hat{\eta}_{11}^k \Psi_i, \Psi_j > \hat{F}_{1c,i}^k + \sum_{i=0}^{N_1} < \hat{\eta}_{12}^k \Psi_i, \Psi_j > \hat{F}_{2c,i}^k + < \hat{\eta}_{13}^k \Psi_i, \Psi_j >$$

$$\sum_{i=0}^{N_1} \hat{A}_{2,i}^k < \Psi_i, \Psi_j > = \sum_{i=0}^{N_1} < \hat{\eta}_{21}^k \Psi_i, \Psi_j > \hat{F}_{1c,i}^k + \sum_{i=0}^{N_1} < \hat{\eta}_{22}^k \Psi_i, \Psi_j > \hat{F}_{2c,i}^k + < \hat{\eta}_{23}^k \Psi_i, \Psi_j >$$

Using the relations developed in Eqs. (16), the above equations are simplified as

$$\hat{A}_{1,i}^k = \frac{1}{h_j^k} \sum_{i=0}^{N_1} < \hat{\eta}_{11}^k \Psi_i, \Psi_j > \hat{F}_{1c,i}^k + \frac{1}{h_j^k} \sum_{i=0}^{N_1} < \hat{\eta}_{12}^k \Psi_i, \Psi_j > \hat{F}_{2c,i}^k + \frac{1}{h_j^k} < \hat{\eta}_{13}^k, \Psi_j >$$

$$\hat{A}_{2,i}^k = \frac{1}{h_j^k} \sum_{i=0}^{N_1} < \hat{\eta}_{21}^k \Psi_i, \Psi_j > \hat{F}_{1c,i}^k + \frac{1}{h_j^k} \sum_{i=0}^{N_1} < \hat{\eta}_{22}^k \Psi_i, \Psi_j > \hat{F}_{2c,i}^k + \frac{1}{h_j^k} < \hat{\eta}_{23}^k, \Psi_j >$$

where $j = 0,...,N_j$. Applying all $j$ projections on the above relations, one can express the projected handle equations of motion of body $k$ as

$$\hat{A}_1^k = \hat{\Gamma}_{11}^k \hat{F}_{1c}^k + \hat{\Gamma}_{12}^k \hat{F}_{2c}^k + \hat{\Gamma}_{13}^k$$

$$\hat{A}_2^k = \hat{\Gamma}_{21}^k \hat{F}_{1c}^k + \hat{\Gamma}_{22}^k \hat{F}_{2c}^k + \hat{\Gamma}_{23}^k$$

Figure 1: Assembling two consecutive bodies to form a new body [16]
In these equations, $\tilde{A}^k_1, \tilde{A}^k_2, \tilde{F}_{1c},$ and $\tilde{F}_{2c}^k \in \mathbb{R}^{6(N_t+1)}$ are column matrices similar to the one defined in Eqn. (17). Further, the $i^j$th $(i = 0, \ldots, N_t$ and $j = 0, \ldots, N_t$) elements of the coefficient matrices i.e., $\tilde{\Gamma}^k_{11}, \tilde{\Gamma}^k_{12}, \tilde{\Gamma}^k_{21},$ and $\tilde{\Gamma}^k_{22}$ are defined as

$$\tilde{\Gamma}^k_{mn,ji} = \frac{1}{H_j^i} < \eta^k_{mn} \Psi_i, \Psi_j>, \quad m = 1, 2, \quad n = 1, 2,$$  

(28)

where each of the above elements is a $6 \times 6$ matrix. Finally, the $j^j$th $(j = 0, \ldots, N_t)$ elements of column matrices $\tilde{\Gamma}^k_{13}$ and $\tilde{\Gamma}^k_{23}$ are expressed as

$$\tilde{\Gamma}^k_{mn,ji} = \frac{1}{H_j^i} < \tilde{\eta}^k_{mn} \Psi_i, \Psi_j>, \quad m = 1, 2,$$  

(29)

while each of the above elements is a $6 \times 1$ column matrix. As such, each of the Eqn. (26) and Eqn. (27) expresses a set of $6(N_t + 1)$ projected handle equations of motion associated with handles $H^k_1$ and $H^k_2,$ respectively, while all of the coefficients in these equations are not symbolically carrying the random variable $\tilde{\xi}$.

5.3 Projected Kinematic Constraint at the Connecting Joint

For a deterministic multibody system, the kinematic constraint at the acceleration level associated with connecting joint $J^k$ between bodies $k - 1$ and $k$, as shown in Fig. 1, is expressed as

$$A^k_1 - A^k_2 = \mathcal{P}^k \dot{u}^k + \mathcal{P}^k u^k.$$  

(30)

In the above relation, the joint-free motion map $\mathcal{P}^k$ [30], its time derivative $\dot{\mathcal{P}}^k$, and its orthogonal complement $\mathcal{D}^k, ((\mathcal{D}^k)^T \mathcal{D}^k = 0),$ are functions of the orientation angles and/or generalized coordinates and/or generalized speeds. For a nondeterministic problem, since states of the system, including generalized coordinates and speeds are projected onto the space of appropriate base functions, these matrices become functions of modal states and uncertain variable $\tilde{\xi}$. As such, they are replaced by, $\mathcal{D}^k, \dot{\mathcal{D}}^k, \ddot{\mathcal{D}}^k$. Therefore, Eq. (30) is modified as

$$\sum_{i=0}^{N_t} (\tilde{A}^k_{1,i} - \tilde{A}^k_{2,i}) \Psi_i = \mathcal{D}^k \sum_{i=0}^{N_t} \tilde{u}^k_i \Psi_i + \dot{\mathcal{D}}^k \sum_{i=0}^{N_t} u^k_i \Psi_i.$$  

(31)

Premultiplying this equation by $(\mathcal{D}^k)^T$, one can eliminate the unknown modal joint acceleration terms $\sum_{i=0}^{N_t} \tilde{u}^k_i \Psi_i$, and arrive at

$$(\mathcal{D}^k)^T \sum_{i=0}^{N_t} (\tilde{A}^k_{1,i} - \tilde{A}^k_{2,i}) \Psi_i = (\mathcal{D}^k)^T \dot{\mathcal{D}}^k \sum_{i=0}^{N_t} u^k_i \Psi_i.$$  

(32)

Applying the Galerkin projection to the above relation yields

$$\sum_{i=0}^{N_t} < (\mathcal{D}^k)^T \Psi_i, \Psi_j > (\tilde{A}^k_{1,i} - \tilde{A}^k_{2,i}) = \sum_{i=0}^{N_t} < (\mathcal{D}^k)^T \dot{\mathcal{D}}^k \Psi_i, \Psi_j > \tilde{u}^k_i, \quad j = 0, \ldots, N_t.$$  

(33)

For a joint which allows $v$ degrees of freedom between bodies $k - 1$ and $k,$ $\mathcal{D}^k$ has the dimension of $6 \times (6 - v)$. As such, this relation shows a set of $6 - v$ equations. Forming the above relation for all values of $j$, and using the $6(N_t + 1) \times 1$ column matrices $\tilde{\mathcal{A}}_1^k$ and $\tilde{\mathcal{A}}_2^k$, one can express the projected constraint equation at joint $J^k$ between bodies $k - 1$ and $k$ as

$$\mathcal{X}^k(\tilde{\mathcal{A}}^k_1 - \tilde{\mathcal{A}}^k_2) = \mathcal{P}^k.$$  

(34)
In this equation, $\tilde{X}^j$ and $\tilde{Y}^j$ are $[(6 - \nu)(N_i + 1)] \times [6(N_i + 1)]$ and $[(6 - \nu)(N_i + 1)] \times 1$ matrices, respectively. The elements of these matrices are obtained by using the following relations

$$\tilde{X}^j_{ij} = \langle (\tilde{\mathcal{G}}^j)^T \Psi_i, \Psi_j \rangle,$$  \hspace{1cm} (35)

$$\tilde{Y}^j_j = \sum_{i=0}^{N_i} \langle (\tilde{\mathcal{G}}^j)^T \tilde{\mathcal{G}}^j, \Psi_i \rangle u^j_i.$$  \hspace{1cm} (36)

It should be mentioned that the elements $\tilde{X}^j_{ij}$ and $\tilde{Y}^j_j$ are $(6 - \nu) \times 6$ and $(6 - \nu) \times 1$ matrices, respectively.

### 5.4 Assembling of Consecutive Bodies

Consider bodies $k - 1$ and $k$ connected together via the kinematic joint $J^k$ as shown in Fig. 1. The projected nondeterministic handle equations of motion of body $k - 1$ are expressed as

$$\tilde{X}^{k-1} = \tilde{R}_1^{k-1} \tilde{F}^{k-1}_{1c} + \tilde{R}_2^{k-1} \tilde{F}^{k-1}_{2c} + \tilde{R}_3^{k-1} \tilde{F}^{k-1}_{13} = \tilde{X}^{k-1}.$$  \hspace{1cm} (37)

$$\tilde{X}^{k} = \tilde{R}_1^k \tilde{F}^{k}_{1c} + \tilde{R}_2^k \tilde{F}^{k}_{2c} + \tilde{R}_3^k \tilde{F}^{k}_{13} = \tilde{X}^{k}.$$  \hspace{1cm} (38)

Replacing Eqns. (24) and (38) in the projected kinematic constraint in Eq. (34), one arrives in the following relation:

$$\tilde{X}^j \tilde{R}_1^k \tilde{F}^k_{1c} + \tilde{X}^j \tilde{R}_2^k \tilde{F}^k_{2c} + \tilde{X}^j \tilde{R}_3^k \tilde{F}^k_{13} - \tilde{X}^j \tilde{R}_1^{k-1} \tilde{F}^{k-1}_{1c} - \tilde{X}^j \tilde{R}_2^{k-1} \tilde{F}^{k-1}_{2c} - \tilde{X}^j \tilde{R}_3^{k-1} \tilde{F}^{k-1}_{13} = \tilde{Y}^j.$$  \hspace{1cm} (39)

The following relation between the stochastic spatial constraint loads at the common joints can be written based on Newton’s third law of motion

$$\tilde{F}^k_{2c} = -\tilde{F}^k_{1c}.$$  \hspace{1cm} (40)

Therefore, Eq. (39) is simplified as

$$\tilde{X}^j (\tilde{R}_1^k + \tilde{R}_2^k) \tilde{F}^k_{1c} = \tilde{X}^j \tilde{R}_1^{k-1} \tilde{F}^{k-1}_{1c} - \tilde{X}^j \tilde{R}_2^{k-1} \tilde{F}^{k-1}_{2c} + \tilde{X}^j \tilde{R}_3^k \tilde{F}^k_{13} - \tilde{X}^j \tilde{R}_3^{k-1} \tilde{F}^{k-1}_{13}.$$  \hspace{1cm} (41)

The mapping $\tilde{\mathcal{G}}^j$ spans the subspace of the constraint load at joint $J^k$. As such, the nondeterministic spatial constraint load $\tilde{F}_{1c}^k \in \mathbb{R}^6$ can be expressed in terms of the column matrix of measure numbers $\tilde{f}_{1c}^k \in \mathbb{R}^{6-\nu}$ as

$$\tilde{F}_{1c}^k = \tilde{\mathcal{G}}^j \tilde{f}_{1c}^k.$$  \hspace{1cm} (42)

Expressing this equation in terms of base functions yields

$$\sum_{j=0}^{N_i} \tilde{F}_{1c,j}^k \Psi_j = \sum_{j=0}^{N_i} \tilde{F}_{1c,j}^k \Psi_j.$$  \hspace{1cm} (43)

Applying the Galerkin projection to the above relation results in

$$\tilde{F}_{1c}^k = \sum_{j=0}^{N_i} \tilde{f}_{1c,j}^k \Psi_j.$$  \hspace{1cm} (44)

Performing the above relation on all of the values of $i$, one can express the mapping between the column matrices $\tilde{F}_{1c}^k \in \mathbb{R}^{6(N_i+1)}$ and $\tilde{f}_{1c}^k \in \mathbb{R}^{(6-\nu)(N_i+1)}$ as

$$\tilde{F}_{1c}^k = \tilde{Z}^j \tilde{f}_{1c}^k.$$  \hspace{1cm} (45)
Therefore, Eqn. (41) is revised as

\[ X^\mu (\tilde{\Gamma}_{11}^k + \tilde{\Gamma}_{22}^{k-1})Z^\mu \tilde{\Gamma}_{ic}^k = X^\mu \tilde{\Gamma}_{21}^{k-1} \tilde{F}_{1c}^{k-1} - X^\mu \tilde{\Gamma}_{12}^{k} \tilde{F}_{2c}^{k} + X^\mu [\tilde{\Gamma}_{23}^{k-1} - \tilde{\Gamma}_{13}^k + \tilde{Y}^\mu]. \]  

(46)

Solving the above equation for \( \tilde{F}_{1c}^k \), one can use the mapping provided in Eq. (45) to find all modal constraint loads \( \tilde{F}_{1c}^k \) and \( \tilde{F}_{2c}^{k-1} \) as

\[ \tilde{F}_{1c}^k = -\tilde{F}_{2c}^{k-1} = \tilde{Z}^\mu \tilde{\Gamma}_{12}^{k-1} \tilde{X}_{21}^{k-1} \tilde{F}_{1c}^{k-1} - \tilde{Z}^\mu \tilde{\Gamma}_{23}^{k-1} \tilde{X}_{22}^{k-1} \tilde{F}_{2c}^k + \tilde{Z}^\mu \tilde{\Gamma}_{13}^{k-1} \tilde{X}_{22}^{k-1} [\tilde{\Gamma}_{23}^{k-1} - \tilde{\Gamma}_{13}^k + \tilde{Y}^\mu], \]  

(47)

where

\[ \tilde{\Gamma} = \tilde{X}^\mu (\tilde{\Gamma}_{11}^k + \tilde{\Gamma}_{22}^{k-1}) \tilde{Z}^\mu. \]  

(48)

This matrix is always invertible since \( \tilde{\Gamma}_{11}^k + \tilde{\Gamma}_{22}^{k-1} \), \( \tilde{X}^\mu \), and \( \tilde{Y}^\mu \) are all full rank matrices. Replacing the expressions for \( \tilde{F}_{1c}^k \) and \( \tilde{F}_{2c}^{k-1} \) from Eq. (47) into Eqs. (37) and (27), one arrives at the following relations

\[ \tilde{\Lambda}_1^{k-1} = \tilde{\Gamma}_{11}^{k-1} \tilde{F}_{1c}^{k-1} + \tilde{\Gamma}_{12}^{k-1} \tilde{F}_{2c}^k + \tilde{\Gamma}_{13}^{k-1}, \]  

(49)

\[ \tilde{\Lambda}_2^k = \tilde{\Gamma}_{21}^{k-1} \tilde{F}_{1c}^{k-1} + \tilde{\Gamma}_{22}^{k-1} \tilde{F}_{2c}^k + \tilde{\Gamma}_{23}^{k-1}, \]  

(50)

where the intermediate variables are defined as

\[ \tilde{\Gamma}_{11}^{k-1} = \tilde{\Gamma}_{11}^{k-1} - \tilde{\Gamma}_{12}^{k-1} \tilde{X}^\mu \tilde{\Gamma}_{21}^{k-1} \tilde{Z}^\mu \]  

(51)

\[ \tilde{\Gamma}_{12}^{k-1} = \tilde{\Gamma}_{12}^{k-1} \tilde{X}^\mu \tilde{\Gamma}_{12}^{k-1} \tilde{Z}^\mu \tilde{\Gamma}_{22}^{k-1} \]  

(52)

\[ \tilde{\Gamma}_{13}^{k-1} = \tilde{\Gamma}_{13}^{k-1} - \tilde{\Gamma}_{12}^{k-1} \tilde{X}^\mu \tilde{\Gamma}_{13}^{k-1} \tilde{Z}^\mu \tilde{\Gamma}_{23}^{k-1} \tilde{Y}^\mu \]  

(53)

\[ \tilde{\Gamma}_{21}^{k-1} = \tilde{\Gamma}_{21}^{k-1} \tilde{X}^\mu \tilde{\Gamma}_{21}^{k-1} \tilde{Z}^\mu \tilde{\Gamma}_{21}^{k-1} \tilde{Y}^\mu \]  

(54)

\[ \tilde{\Gamma}_{22}^{k-1} = \tilde{\Gamma}_{22}^{k-1} - \tilde{\Gamma}_{21}^{k-1} \tilde{X}^\mu \tilde{\Gamma}_{22}^{k-1} \tilde{Z}^\mu \tilde{\Gamma}_{22}^{k-1} \tilde{Y}^\mu \]  

(55)

\[ \tilde{\Gamma}_{23}^{k-1} = \tilde{\Gamma}_{23}^{k-1} + \tilde{\Gamma}_{21}^{k-1} \tilde{X}^\mu \tilde{\Gamma}_{23}^{k-1} \tilde{Z}^\mu \tilde{\Gamma}_{23}^{k-1} \tilde{Y}^\mu \]  

(56)

It is observed that Eqns. (49) and (50) have the same form as Eqns. (20) and (21) for the projected handle equations of a single body. In other words, one can combine the adjacent bodies and arrive at the stochastic handle equations of the assembly \( k - 1 : k \) as shown in Fig. 1. In order to form and solve the PCE-based equations of motion of a non-deterministic multibody system in the DCA framework, one should perform a series of assemblies and disassemblies as described in the following section.

### 5.5 Implementation: Assembly and Disassembly Passes

In the assembly pass, larger encompassing subsystems are recursively generated by assembling adjacent articulated bodies/subsystems of a multibody problem as shown in Fig. 1. The process starts at the body (leaf) level of the binary tree by constructing the projected handle equations of motion of each individual body. The resulting projected handle equations of consecutive bodies \( k - 1 \) and \( k \) are then combined together as described in section 5.4 to compute the projected handle equations of the assembly \( k - 1 : k \). Using the presented recursive formulas, the projected handle equations of motion of the generated subassemblies are then used to produce the equations associated with the resulting assemblies. This process is then implemented hierarchically by coupling together pairs of adjacent subassemblies, and forming new assemblies, as shown in Fig. 2, to express the entire articulated system with \( n_b \) bodies in terms of the projected handle equations of motion of a single all encompassing assembly (root node) as

\[ \tilde{\Lambda}_1^1 = \tilde{\Gamma}_{11}^{1,1} \tilde{F}_{1c}^1 + \tilde{\Gamma}_{12}^{1,1} \tilde{F}_{2c}^1 + \tilde{\Gamma}_{13}^{1,1}, \]  

(57)

\[ \tilde{\Lambda}_2^{n_b} = \tilde{\Gamma}_{21}^{1,n_b} \tilde{F}_{1c}^1 + \tilde{\Gamma}_{22}^{1,n_b} \tilde{F}_{2c}^1 + \tilde{\Gamma}_{23}^{1,n_b}. \]  

(58)
The disassembly process is initiated by applying the boundary conditions to the stochastic equations of the root node of the binary tree i.e. Eqns. (57) and (58). These equations are then solved for the unknown modal constraint loads and/or accelerations of the terminal handles of the entire system as indicated in Fig. 2. These known values are then substituted into the projected handle equations of the associated subassemblies to evaluate modal constraint loads and accelerations at common handles of subassemblies. This process is repeated in a hierarchic disassembly of the binary tree so that at the end of the disassembly process, all modal accelerations and/or time derivative of the modal generalized speeds, as well as modal constraint loads of all bodies of the system are determined. In the serial and parallel implementations, this method reduces the computational cost of forming and solving the equations of motion to \(O(n(N_t + 1))\) and \(O(\log(n(N_t + 1)))\) which is a significant improvement over the traditional methods with the computational complexity of \(O(n(N_t + 1))^3\).

### 6 SIMULATION RESULTS

Herein, the simulation results are provided based on the work performed in [16], where the random variables are carried through the DCA process, to demonstrate how PCE can capture the system dynamics in comparison to Monte Carlo simulations. In this example, a double pendulum problem shown in Fig. 3 is considered with uniform bars of the unit mass and the length of two units. In this example, which is an extension of the one presented in [2] for a single pendulum, uncertainty with uniform distribution \(L_2 = 1 + 0.2\zeta\) (Units) \((\zeta \in [-1, 1])\) exists in the location of the mass center of the second pendulum which indicates that \(L_2 \in [0.8, 1.2]\) (Units). Legendre polynomials up to the first, second, and third orders have been used to study the dynamic behavior of this system. The PCE-based DCA simulations have been performed for 10s with the initial conditions \(q_1(0) = 0.1\) rad, \(q_2(0) = 0\) rad, and \(u_1(0) = u_2(0) = 0\) rad/s. Figure 4 demonstrate the mean and standard deviation of \(q_1\) and \(u_1\). It is demonstrated that these represented stochastic quantities from PCE are in a relatively good agreement with that of 2000 MC simulations. Indeed, the order
Figure 4: Comparing the mean and standard deviation of $q_1$ and $u_1$ from 2000 MC simulations with those from the first, second and third order PCE simulations [16]

of the PCE affects the accuracy of higher order expectation values.

7 CONCLUSIONS

Herein, the method of Divide-and-Conquer Algorithm has been generalized to formulate and solve the equations of motion of a multibody system under the influence of the uncertainty. In this scheme, uncertainty is modeled as the polynomial chaos expansion (PCE) of the state variables of the system. The fundamental formulations to construct the PCE method for multibody systems have been presented. Further, Galerkin projection to compute the projected handle equations of motion of the system has been developed. The projected constraint equation between the consecutive bodies at the acceleration level has been derived. Unlike the previous techniques, in this derivation, random variables do not symbolically appear in the equations of motion and constraint equations. The mathematical framework to compute the handle equations of the assembly of consecutive bodies/assemblies has been presented. The projected DCA works using the assembly and disassembly sweeps. In the assembly pass, the projected handle equations of motion of the adjacent bodies and subassemblies are coupled together to form the nondeterministic handle equations of the resulting assemblies. In the disassembly pass, the resulting equations of motion of subassemblies are recursively solved for the modal spatial accelerations and spatial constraint loads. The proposed method is highly parallelizable and scales down the computational complexity as a linear and logarithmic function of the state variables in serial and parallel implementations, respectively.
REFERENCES


Multibody model reduction by parameter elimination

Xabier Iriarte*, Javier Ros*, Vicente Mata#, Aitor Plaza*

* Mechanical, Energy and Materials Engineering Department
Public University of Navarre
Arrosadia Campus, s/n, 31006, Pamplona-Iruñea, Spain
[xbier.iriarte, jros, aitor.plaza]@unavarra.es

# Centro de Investigación de Tecnología de Vehículos, Universidad Politécnica de Valencia
C/Camino de Vera s/n, 46022 Valencia, Spain
vmata@mcm.upv.es

ABSTRACT
Lots of different techniques are currently used for the reduction of multibody system models. However, model reduction by parameter elimination apparently has not yet been used in this field. As the multibody inverse dynamic models can be written linearly with respect to the inertial parameters, it looks reasonable to adapt the model selection techniques used in statistics to reduce dynamic models by parameter elimination.

In this paper a parameter-elimination-based model reduction technique is introduced. This technique can control the accuracy of the reduced method for a characteristic operational regime that can be defined in different ways.

This technique has been applied to two 6 degrees of freedom robots: an open-loop Puma and a closed-loop Hexaglide. Very good results have been obtained, reducing the computational time as far as a 40%.

Keywords: Model Reduction, Parameter Elimination, Base Parameters, Model Selection, Parameter Estimation.

1 Introduction
Multibody models are usually computationally expensive and doing efficient simulations or reaching the real-time computation usually requires some sort of model reduction or simplification. This has made of model reduction a fundamental topic in the MBS bibliography.

There exist several ways in which multibody models can be reduced. Linearization, Modal Approximation, Condensation, Component Mode Synthesis, Krylov Subspaces, SVD-based techniques or the Proper Orthogonal Decomposition are some of these reduction techniques. For a brief survey on these techniques see [13].

One model reduction technique that is not broadly used in the context of Multibody Systems is that of eliminating parameters of the model. Taking advantage of the linearity of the equations of motion with respect to the inertial parameters [6], some parameters can be neglected so that the nonlinear functions that appear multiplying them do not have to be evaluated, and thus, the model is reduced.

Great effort has been done in the field of statistics and system identification to find the best model that explains a data vector as a linear combination of some regressors [9]. This determination is a combination of two tasks: the selection of a criterion to decide which of two models is best [20], and an algorithm to efficiently evaluate the selected criterion for all the possible parameter combinations [2, 9, 3, 11]. When sufficient data are available, cross validation techniques can be used, which use the prediction error of the validation data as the model selection criterion. When not many data are available, some of the classic criteria for the best model selection are $C_p$, FPE, AIC, BIC, and MDL [8, 10, 17].

These model selection techniques are closely related to model reduction (since they obtain a model with a reduced number of regressors) but their approach is statistical. They assume that the output
variables are measured in an experiment and the best model is, for instance, the one that minimizes the expectation of the prediction error (FPE criterion). However, in the model reduction approach, the goal is to obtain the most efficient model with a prediction error smaller than a tolerance.

In this paper a multibody model reduction technique is presented which is closely related to the model selection techniques developed in statistics and system identification. The reduction technique is based on the linearity of the equations of motion with respect to the inertial and other parameters. It carefully selects the parameters to be eliminated so that the norm of the prediction error of the generalized forces does not exceed a certain tolerance.

2 Model Reduction Method

2.1 The Inverse Dynamic Model

The Lagrange equations of a multibody system can be written as

$$M(q, \dot{q})\ddot{q} - \delta(q, \dot{q}) = \tau$$

where $q$, $\dot{q}$ and $\ddot{q}$ are the generalized coordinates, velocities and accelerations, $M$ represents the mass matrix, $\tau$ the vector of external forces, and $\delta$ represents the Coriolis, centrifugal and constitutive forces. If the Inverse Dynamics Model (IDM) is desired, it is customary to write the equations as

$$K(q, \dot{q}, \ddot{q})\phi = \tau$$

where $K$ represents a one instant observation matrix that depends on the generalized coordinates and their derivatives. The inertial parameters of the solids will be part of vector $\phi$, as well as the parameters used to model springs, dampers, and friction models. The goal of the model reduction by parameter elimination is to eliminate some columns of $K$ (and the corresponding parameters of $\phi$) so that vector $\tau$ can be calculated, in terms of $q$, $\dot{q}$ and $\ddot{q}$, as a linear combination of some of the columns of $K$ as:

$$K_R(q, \dot{q}, \ddot{q})\phi_R \approx K(q, \dot{q}, \ddot{q})\phi = \tau.$$

The computational cost of calculating $\tau$ decreases as $K$ and $\phi$ are reduced to $K_R$ and $\phi_R$. Some authors [12, 14] suggest to directly eliminate some of the parameters with small relative values of mass or inertia, or those related to Coriolis terms, hoping that their influence in the forces will be negligible. However, there is no systematic criterion to decide if a parameter should be eliminated or not, while depending on the kinematics of the system, the parameter should be relevant or not.

2.2 The Base Parameter Reduction

When one needs to identify the dynamic parameters of a robotic system, it is always necessary to re-parameterize the model in order to assure the identifiability of the model parameters. This procedure leads to a model which depends on a smaller number of parameters so that the re-parameterization can be interpreted as a model reduction technique. This reduction can be performed numerically [4] as follows.

When writing the inverse dynamics equations of a multibody system for a set of $n$ instants, collecting all of them, the Observation Matrix ($W$) is obtained as:

$$W\phi = \begin{bmatrix} K(q_1, \dot{q}_1, \ddot{q}_1) \\ \vdots \\ K(q_n, \dot{q}_n, \ddot{q}_n) \end{bmatrix} \phi = \begin{bmatrix} \tau_1 \\ \vdots \\ \tau_n \end{bmatrix} = \chi.$$

In order to obtain $n$ different instants to build matrix $W$, a trajectory is usually design. All the poses of this trajectory must belong to the workspace of the system and the trajectory is usually optimized in order to be permanently exciting [18, 19].
It turns out that for a general set of instants, matrix $W$ tends to be rank deficient meaning that linear dependencies between its columns exist. Thus, writing $W = [W_1, W_2]$ and $\phi' = (\phi_1, \phi_2')$, some of the columns of $W$ can be written as linear combinations of the others

$$W_2 = W_1 \beta$$  \hspace{1cm} (1)

and the reduced model can be calculated as

$$W\phi = [W_1, W_2] \{\phi_1\} = [W_1, W_1\beta] \{\phi_1\} = W_1[I, \beta] \{\phi_1\} = W_1(\phi_1 + \beta \phi_2) = W_b \phi_b.$$  \hspace{1cm} (2)

Therefore, a set of independent columns of $W$ will constitute the reduced model observation matrix, $W_b$, while the reduced parameter vector, $\phi_b$, the so called base parameter vector, can be written as a linear combination of the original parameters:

$$\phi_b = \phi_1 + \beta \phi_2.$$  \hspace{1cm} (3)

Thus, if Equation (1) perfectly holds, Equation (2) will represent a model reduction without any kind of approximation.

If the model reduction is performed by simply eliminating $W_2 \phi_2$ from Equation (2), the remaining parameters will represent the so called minimum parameters. In this situation, while the model observation matrix will be the same as in the base parameters approach, the physical sense of the parameters will be lost. Let us therefore note that the reduced model in terms of base parameters actually includes the information of the eliminated parameters. Advantage of this property will be taken when the more general model reduction method is described.

The kinematic couplings of the joints in a multibody system are the responsible for the linear combinations (rank deficiency) of matrix $W$. In fact, the base parameter expressions of Equation (3) can be obtained symbolically [5, 15] with the sole information of the kinematics of the system. However, for a number of reasons, some nearly linear combinations of the columns of $W$ can exist, and treating those columns as if they were perfect linear combinations of the other, the model can be further reduced. Unlike in Equation (2), in this model reduction approach a simplification of the model will be done.

### 2.3 Core of the model reduction algorithms

Let us suppose that our current model is $W\phi = \chi$. Let us denote $w_i$ the $i^{th}$ column of $W$ and let $W_i$ be matrix $W$ without its $i^{th}$ column. Equivalently, $\phi_i$ represents vector $\phi$ without its $i^{th}$ element. In the model reduction algorithm, a column vector $w_i$ is eliminated from $W$ (and the corresponding parameter $\phi_i$ from $\phi$) if the norm of the part of $w_i\phi$ that cannot be written as a linear combination of the columns of $W_i$ is smaller than a tolerance. Mathematically, vector $w_i$ could be eliminated from $W$ if

$$\frac{\|\chi - W_i \hat{\phi}_i\|}{\|\chi\|} < tol.$$  \hspace{1cm} (4)

where $\hat{\phi}_i = W_i^+ \chi$ and $W_i^+$ represents the pseudo-inverse matrix of $W_i$.

If vector $w_i$ was eliminated from $W$, part of the information of vector $w_i$ could still be included in the reduced model. Substituting vector $w_i$ by its projection onto the space of the columns of $W_i$,

$$w_i \approx w_i' = W_i W_i^+ w_i$$

vector $w_i'$ could be written as a linear combination of the columns of $W_i$ as

$$w_i' = W_i \beta$$
and the model could be reduced as

$$W_{\phi} = [W_i, w_i] \left\{ \phi_i \right\} \approx [W_i, W_i \beta] \left\{ \phi_i \right\} = W_i [I_1, \beta] \left\{ \phi_i \right\} = W_i (\phi_i + \beta \phi_i) = W_R \phi_R.$$  \hspace{1cm} (5)

Based on this approximations two algorithms are proposed in this paper for a multibody model reduction.

### 2.4 The backward elimination algorithm

Starting from the full multibody model $W_{\phi} = \chi$ with $p$ parameters, the magnitude $e_i$ is calculated for the $p$ models that can be built eliminating, in each of them, the $i^{th}$ column of $W$, being

$$e_i = \frac{||\chi - W_i \tilde{\phi}_i||}{||\chi||}.$$  \hspace{1cm} (6)

where $\tilde{\phi}_i = W_i^+ \chi$. Calling $e_j$ the smallest of all $e_i$ (for $i = 1, \ldots, p$) if $e_j < tol$, $w_j$ will be eliminated from $W$ and $\phi_j$ will be eliminated from $\phi$. For the next iteration of the algorithm $W_j$, $\phi_j$ and $p - 1$ will play the role of $W, \phi$ and $p$, respectively. The algorithm will stop eliminating parameters when $e_j \geq tol$.

This algorithm for determining the parameters that will be eliminated is similar to the backward elimination algorithm used in model selection [2, 7, 9]. However, the error $e_i$ and the stopping criterion are different.

Once the parameters to be eliminated are selected, the matrix of the full model can be divided into two submatrices $W_1$ and $W_2$ so that $W = [W_1, W_2]$ and $W_2$ represents the eliminated columns. Approximating the columns of $W_2$ by their projection onto the space of the columns of $W_1$,

$$W_2 \approx W_1 W_1^+ W_2 = W_1 \beta$$

the model can be reduced as:

$$W_{\phi} = [W_1, W_2] \left\{ \phi_1 \right\} \approx [W_1, W_1 \beta] \left\{ \phi_1 \right\} = W_1 [I_1, \beta] \left\{ \phi_1 \right\} = W_1 (\phi_1 + \beta \phi_2) = W_R \phi_R.$$  \hspace{1cm} (7)

where $\phi_R = \phi_1 + \beta \phi_2$ can be thought as the generalized base parameters of the reduced model. The explicit expressions for $\phi_R$ provide valuable information on the system and the way in which the parameters are grouped for reducing the model.

### 2.5 The forward selection algorithm

Starting without any column of $W$ the magnitude $e_i$ is calculated for the $p$ models that can be built with the $i^{th}$ column of $W$, being

$$e_i = \frac{||\chi - W_i \tilde{\phi}_i||}{||\chi||},$$  \hspace{1cm} (8)

where $\tilde{\phi}_i = W_i^+ \chi$. Matrix $W_j$ represents the current $W$ matrix (void for the first iteration) to which the vector $w_j$ of the full model has been added. Calling $e_j$ the smallest of all $e_i$ (for $i = 1, \ldots, p$) if $e_j > tol$, $w_j$ of the full model will be added to the current $W$ and $\phi_j$ will be added to the current $\phi$. For the next iteration of the algorithm $W_j, \phi_j$ and $p + 1$ will play the role of $W, \phi$ and $p$. The algorithm will stop adding parameters to the model when $e_j \leq tol$ or when all the parameters of the full model have been added.

This algorithm for determining the parameters that will be added is similar to the forward selection algorithm used in model selection [2, 7, 9]. However, the error $e_i$ and the stopping criterion are different.

As in the backward elimination algorithm, once the parameters to be added are selected, the matrix of the full model is divided into two submatrices and the model reduction is performed as in Equation (7).
2.6 The QR based algorithm

An alternative algorithm based on the QR decomposition of the full model matrix \( W \) can also be proposed. This algorithm serves as a reference for comparing the backward elimination and forward selection algorithms since it uses a standard QR decomposition and automatically sorts the parameters in order of relevance.

Let \( Q, R \) and \( E \) be three matrices so that

\[
WE = QR
\]

where \( Q \) is a unitary matrix, \( R \) is upper triangular and \( E \) is a permutation matrix so that the absolute value of the diagonal elements of \( R \) are decreasing. As noted by Gautier [4], if the last \( p - r \) diagonal elements of \( R \) are null, the matrix product \( WE \) can be written as:

\[
WE = [W_1, W_2] = [Q_1, Q_2] \begin{bmatrix} R_1 & R_2 \\ 0_{(p-r)\times r} & 0_{(p-r)\times (p-r)} \end{bmatrix} = [Q_1 R_1, Q_1 R_2]
\]

where \( R_1 \) is a \( r \times r \) regular matrix. Then it comes:

\[
W_1 = Q_1 R_1 \quad \text{and} \quad W_2 = Q_1 R_2
\]

from which we deduce the relation

\[
W_2 = W_1 \beta
\]

where \( \beta = R_1^{-1} R_2 \). Without any approximation, for perfectly null values of the last \( p - r \) diagonal elements of \( \hat{R} \), the model could be reduced as in Equation (2). However, if the values of the last diagonal elements of \( R_1 \) are small, the model can further be reduced (with an approximation) passing the last column of \( R_1 \) to \( R_2 \) and reducing the value of \( r \) by one.

As the diagonal elements of \( R \) are ordered by the QR decomposition so that the first columns of \( R \) are the most relevant, eliminating one by one the last columns of \( R_1 \) and calculating the error of the model calculating \( \chi \), the model can be further reduced. Denoting by \( R_1 \) the matrix \( R \) in which the last \( p - r \) columns have been substituted by zeros, and being \( W_f = QR_f \hat{E} \), the normalized prediction error of the reduced model \( e_r \) could be written as:

\[
e_r = \frac{\| \chi - W_f \phi_f \|}{\| \chi \|} \tag{9}
\]

where \( \phi_f = W_f^{-1} \chi \). The model reduction algorithm would eliminate columns, one by one from last to first, until \( e_r \geq \text{tol} \).

Rewriting \( WE \) for the reduced model with \( p - r \) parameters

\[
WE = [W_1, W_2] = [Q_1, Q_2] \begin{bmatrix} R_{11} & R_{12} \\ 0_{(p-r)\times r} & R_{22} \end{bmatrix} = [Q_1 R_{11}, Q_1 R_{12} + Q_2 R_{22}].
\]

Then, it comes

\[
W_1 = Q_1 R_{11} \quad \text{and} \quad W_2 = Q_1 R_{12} + Q_2 R_{22}.
\]

Approximating the columns of \( W_2 \) by their projection onto the space of the columns of \( W_1 \),

\[
W_2 \approx W_2^* = W_1 W_1^* W_2 = W_1 R_{11}^{-1} R_{12} = W_1 \beta.
\]

So matrix \( W_2^* \) could be written as a linear combination of the columns of \( W_1 \).

Therefore, for the approximated case, using \( W_1 \) and \( W_2^* = W_1 \beta \), the reduced model would be calculated as

\[
W \phi = (WE)(E \phi) = [W_1, W_2] \{ \phi_1 \phi_2 \} \approx [W_1, W_1 \beta \{ \phi_1 \phi_2 \} = W_1 (\phi_1 + \beta \phi_2) = W R \phi_R.
\]
3 Results

The proposed model reduction methods have been applied to two different 6 DOF robotic systems: a Puma open-loop robot [1] and a Hexaglide closed-loop robot [16]. A classic validation procedure has been used where one trajectory is used for model reduction while another one is used for validation purpose. The models have been reduced using exciting trajectories [18, 19]. This trajectories have been optimized to minimize the condition number of matrix $W_p$ in Equation (2).

As the procedures eliminate or add one parameter at a time, all the intermediate models have been saved, each of them with a different number of parameters. For each of them, the prediction error has been calculated in order to observe how it increases as the models depend on less parameters.

The trajectories used for validation have been selected to be as demanding (or slightly less demanding) than those used for model reduction. In this way the reduced models will be also valid to accurately predict the output of different trajectories than that used for the reduction. Several trajectories have been obtained as local minima of the same optimization procedure and a couple of those trajectories that had similar values of the objective function have been selected for the analysis, choosing the reduction trajectory to be the most demanding of the two.

The model reduction and validation procedure has been done as follows. The observation matrix is evaluated with the reduction trajectory data and the model reduction is performed, obtaining a $p$ parameters reduced model $(W_p)$. The calculation of the values of the reduced model parameters can be done as $\Phi_p = W_p^+\chi_{red}$, where $W_p^+$ is the pseudo-inverse of matrix $W_p$ evaluated with the reduction trajectory data, and $\chi_{red}$ is the $\chi$ vector of the reduction trajectory. In this situation, the validation of the reduced models is performed calculating the next error:

$$e_p = \frac{||\chi_{val} - W_{p, val} \Phi_p||}{||\chi_{val}||} \quad (10)$$

This way, the parameters estimated in the reduction process have been used for the validation.

As the reduced model is devoted to calculate the forces within a tolerance, the error criteria of Equations (6), (8) and (9) have been calculated with the $\infty$-norm ($||v||_\infty = \max (v)$). This norm can be more interesting for a designer than the classic 2-norm if the maximum error for an instant is desired to be bounded by $tol$ rather than the Mean Square Error of a trajectory. However, the same procedure can be used with the classic 2-norm while the results behave in a similar fashion.

The results obtained applying these algorithms to the Puma and Hexaglide robots are the ones in Figures (1) and (2), respectively. The solid lines represent the error of the reduced model calculating the forces of the reduction trajectories. The dashed lines represent the errors of the reduced models (obtained using the reduction trajectories) calculating the forces of the validation trajectories. The three different colors (red, green and blue) denote the three different reduction methods presented in this paper (backward elimination, forward selection and QR decomposition).

4 Discussion

One of the most important characteristics of the reduction methods is that the model designer can choose the desired trade-off between prediction precision and computational cost. In this way, for relatively small precision requirements, the model can be reduced to a very small amount of parameters, obtaining a very efficient and still accurate reduced model. Moreover, if the required accuracy is that of the machine precision, the models can still be significantly reduced.

As the proposed reduced models are different combinations of some of the columns of the observation matrix, if a reduced model of $r$ parameters is desired while the full model has $p$ parameters, the number of possible different combinations is $\binom{p}{r}$. If all the possible models from 1 to $r$ parameters are desired, $2^p$ possible combinations will have to be evaluated. For moderate $p$ and $r$ it is impossible to evaluate the prediction error of all the possible combinations to find the most
The relative error criteria (normalization dividing by $\|\tau\|$) provides a dimensionless criteria which makes it easy to decide the desired level of prediction error. A relative error level of $10^{-4}$ should be sufficient for most practical applications.

Observing Figures (1) and (2) it is worth noting that the number of parameters needed to obtain a prediction error below $10^{-12}$ has been reduced from 64 (the number of base parameters) to 39 in the Hexaglide robot, and from 36 (number of base parameters) to 20 in the Puma. In both reductions the backward elimination method has been used since it shows to be the most effective of the three. The computational cost should be decreased (if the computational cost of the evaluation is significant) and the accuracy of the model should be increased.
Figure 2. Relative Prediction Error for the Hexaglide robot. Solid lines represent the error of the reduction trajectory data, and dashed lines represent the error of the validation trajectory data.

of each column of $W$ was equivalent) in a 39% in the hexaglide and in a 44% in the puma. Both Figures show that, if a bigger prediction error is allowed (say $10^{-4}$) the models can further be reduced to a very small amount of parameters alleviating even further the computational cost.

REFERENCES


Parallel recursive Hamiltonian formulation for constrained multibody system dynamics

Krzysztof Chadaj, Paweł Malczyk, Janusz Frączek

Division of Theory of Machines and Robots
Institute of Aeronautics and Applied Mechanics
Faculty of Power and Aeronautical Engineering
Warsaw University of Technology
Nowowiejska 24, 00-665 Warsaw, Poland
[kchadaj, pmalczyk, jfraczek]@meil.pw.edu.pl

ABSTRACT
This paper presents a novel recursive divide and conquer formulation for the simulation of complex constrained multibody system dynamics based on the Hamilton’s canonical equations (HDCA). The systems under consideration are subjected to holonomic constraints and may include serial chains, tree chains or closed-loop topologies. Although the Hamilton’s canonical equations exhibit many advantageous features compared to their acceleration based counterparts, it appears that there is a lack of dedicated parallel algorithms for multi–rigid body system dynamics based on the Hamiltonian formulation. The developed HDCA formulation leads to a two-stage procedure. In the first phase, the approach utilizes the divide and conquer scheme, i.e. a hierarchic assembly-disassembly process by traversing the multibody system topology in a binary tree manner to evaluate the joint velocities and constraint impulsive loads. The process exhibits linear $O(n)$ ($n$ – number of bodies) and logarithmic $O(\log_2 n)$ numerical cost, in serial and parallel implementations, respectively. The time derivatives of the total momenta are directly evaluated in the second parallelizable step of the algorithm. Sample closed-loop test cases indicate very well constraint satisfaction at the position and velocity level as well as marginal energy drift without any additional form of constraint stabilization techniques involved in the solution process. The results are comparatively set against more standard acceleration based Featherstone’s DCA approach to indicate the performance of the HDCA algorithm.

Keywords: Forward dynamics, closed–loop systems, Hamilton’s canonical equations, parallel algorithm, divide and conquer.

1 INTRODUCTION
1.1 Literature review
With the increase of computing capabilities, the importance of efficient and real–time simulations of mechanical systems has significantly increased. More and more complex industrial problems encountered in e.g. automotive, biomechanical or robotics areas require faster, more efficient and accurate approaches. With the widespread use of multi–core architectures and general–purpose graphics processing units (GPGPU), more attention has been devoted to algorithms, which may be adopted for parallel implementations [1]. Multibody dynamics recursive algorithms of linear $O(n)$ order become a starting point for further considerations [2–4]. Some of the authors started to work on completely new algorithms [5–8]. Current schemes predominantly depend on the divide and conquer formulation developed by Featherstone [9] and extended by the others.

The most efficient rigid–body dynamics algorithms encountered in the literature are formulated on the basis of Newton–Euler’s equations of motion. On the other hand, the Hamilton’s canonical equations, in which the state of the system is described in terms of generalized coordinates and conjugated canonical momenta, constitute an interesting alternative. One can find multiple applications of canonical equations in such areas as statistical mechanics, quantum mechanics or
molecular dynamics simulations. As known from the classical mechanical textbooks [10] or papers [11], and from modern course books as well [12–14] the Hamilton’s equations present more profitable characteristics compared to their analogous counterparts at the acceleration level. As this fact has not been overlooked, a few articles have been developed on the application of the Hamilton’s equations of motion in the field of analysis of general and constrained multibody systems [15]. Some of the authors noticed that the Hamiltonian formalism provides a handy tool for investigation of intermittent motion with possible impact and collision events [16]. In [17], the comparison between the mentioned formulation and Newton-Euler based algorithm reveals the predominance of the former approach. Other interesting application of the canonical equations can found in [18,19], where the authors developed recursive \( O(n) \) computational complexity algorithms for multi-rigid body dynamics. Initially, the formulations are elaborated for serial chains, and, subsequently extended to deal with closed-loop topologies. Nevertheless, the approach is purely sequential, as it is based on the idea of articulated–body inertias.

Surprisingly, the literature review reveals that it is difficult to find a fully parallel approach for multi-rigid body dynamics simulation based purely on the Hamiltonian formulation. The probable reason for that is the difficulty in adaptation of the classical formulation in an efficient, parallel manner. The authors would like to present a divide and conquer algorithm based on the Hamilton’s canonical equations, which will be abbreviated in the text as HDCA. In this paper, the algorithm’s formulation is adopted to the closed loop topology and competitively set against analogous DCA method results.

1.2 Hamilton’s canonical equations
The state of considered multibody system having \( n \) degrees of freedom can be described by a set of first–order Hamilton’s differential equations in the form

\[
\dot{q} = \left( \frac{\partial \mathcal{H} (t, q, p)}{\partial p} \right)^T \tag{1}
\]

\[
\dot{p} = -\left( \frac{\partial \mathcal{H} (t, q, p)}{\partial q} \right)^T + Q - \Phi^T \lambda \tag{2}
\]

\[
\Phi (t, q) = 0 \Rightarrow \Phi (t, q, \dot{q}) = \Phi_q \dot{q} + \Phi_t = 0 \tag{3}
\]

where \( q, p \) are the system’s redundant canonical coordinates and momenta, \( Q \) are external, applied forces, \( \Phi \) are \( m \) algebraic constraint equations, which have to be fulfilled, and which are associated with unknown Lagrange multipliers \( \lambda \). In case of mechanical system subjected to holonomic constraints, the Hamiltonian \( \mathcal{H} \) is a sum of potential \( V \) and kinetic \( T \) energy. The Hamilton’s canonical equations can be slightly reformulated in order to obtain a form, which is more suitable for numerical calculations [16]:

\[
p = M \dot{q} - \Phi^T_q \sigma \tag{4}
\]

\[
\dot{p} = -V^T_q - \Phi^T \sigma \tag{5}
\]

where \( \sigma \) are the unknown constraint impulsive Lagrange multipliers associated with the constraints equations at the velocity level. Equations (3), (4) and (5) constitute a set of \( 2n + m \) equations that need to be solved and integrated to calculate the state of the system in the next time instant. To be able to solve the above equations, the initial canonical coordinates and momenta are required to be known. As the velocity terms \( \dot{q} \) are easier to measure and operate, the momenta \( p \) can be calculated by using the equivalent equation (4). One have to assume that initially no momentum is absorbed by joints and all impulsive reaction forces are known to be zero (see [16]).
the main and basic concept was widely described in terms of open-loop topologies. In this paper the HDCA algorithm is adopted to mechanisms containing closed-loops as well. Although some ideas remain the same, there are a few new interesting features used to handle kinematic loops. The method solves Hamilton’s canonical equations for constrained multi-rigid body system in a divide and conquer manner allowing easy and efficient implementations on parallel computing architectures. While the state of the mechanical system is described in terms of joint coordinates and momenta, the absolute coordinates are used as well to clearly present the underlying relationships between variables. In the first step joint velocities are computed in the assembly-disassembly scheme along with all impulsive constraints forces using momentum conservation principle for articulated momenta and the superposition principle for impulsive forces. Afterwards, the equations of motion are expressed in terms of joint’s articulated momenta and articulated external loads. Lagrange theorem is used to include the influence of redundant constraints on the mechanism’s motion. For parallel implementation, the algorithm exhibits logarithmic \( O(\log_2 n) \) numerical cost in terms of \( n \) bodies in the system and \( n \) computational threads available. For sequential programming the cost remains linear \( O(n) \).

### 2.1 Joint velocities and impulsive constraint forces

**Articulated momentum definition and conservation principle**

At an arbitrary point \( O \), which is not coincident with body’s center of mass \( C \), the spatial six-dimensional momentum vector for a general body can be defined in a matrix form as \( P_O \in \mathbb{R}^6 \) [18]

\[
P_O = M_O V_O = \begin{bmatrix} mI & -m\omega_C \\ m\omega_C & J_O \end{bmatrix} \begin{bmatrix} v_O \\ \omega \end{bmatrix}
\]  

(6)

In equation (6), all expressions are derived with respect to the point \( O \) and inertial reference frame. The term \( M_O \in \mathbb{R}^{6 \times 6} \) describes the mass matrix of the body for a given vector \( \omega_C \) between points \( O \) and \( C \), while \( V_O \) is a spatial velocity vector composed of translational \( v_O \) and rotational \( \omega \) components. The term \( m \) is the mass of the body and \( J_O \) is the moment of inertia. The identity matrix is defined as \( I \in \mathbb{R}^{3} \) and the tilde symbol above the vector defines a skew-symmetric matrix associated with a given vector.

Let us consider a generic or compound rigid body \( A \) connected with the rest of the multibody system by general kinematic joints at boundary points \( O_1 \) and \( O_2 \) as depicted in Figure 1. From now on, let us consider such points as body’s interfaces called handles (see [3]), by means of which the body communicates with the rest of the system using both active and reaction force’s components. The definition of the handle can be extended onto the velocity level by considering spatial impulsive forces, both active and reaction components, and momentum vectors as well. Subsequently, the spatial articulated momentum vector \( \dot{P} \in \mathbb{R}^6 \) of linear and angular momenta related to the handle can be defined as

\[
\dot{P} = D\sigma + Hp
\]

(7)

where matrix \( H \in \mathbb{R}^{6 \times n_f} \) represents the joint’s motion subspace (\( n_f \)—degrees of freedom for the joint), whereas \( D \in \mathbb{R}^{6 \times (6-n_f)} \) is associated with the joint’s constrained directions. Both subspaces...
are orthogonal to each other, which yield the condition

$$D^T H = 0$$

Equation (7) assumes that the articulated momentum vector taken up from the system can be decomposed into the parts related to the active component $Hp$, and impulsive reaction force component $T = D\sigma$. Vector $p \in \mathbb{R}^{n_f \times 1}$ denotes the joint canonical momenta, whereas vector $\sigma \in \mathbb{R}^{6-n_f}$ indicates constraint impulsive loads (Lagrange multipliers at the velocity level) at joints. The derivative of Lagrange multipliers at the velocity level is equal to the Lagrange multipliers at the acceleration level ($\dot{\sigma} = \lambda$). For an exemplary body $A$ (see Fig. 1) with two handles at points $O_1$ and $O_2$, the momentum conservation principle can be formulated with the use of articulated momentum vectors in form

$$M_{A1}^A \dot{V}_{A1} = P_{A1}^A + S_{12}^A P_{A2}^A$$

$$M_{A2}^A \dot{V}_{A2} = S_{21}^A P_{A1}^A + P_{A2}^A$$

where the subscripts 1 or 2 indicate the $O_1$ or $O_2$ points of application. The above equations make use of the shift matrices $S_{12}^A$ and $S_{21}^A$, which are a handy tool for the transformation of six-dimensional momentum, force and velocity vectors to another point of operation [1]. For the position vector $l_{12}$ measured from the handle point $O_1$ to $O_2$, the shift matrix has the form

$$S_{12}^A = \begin{bmatrix} I & 0 \\ l_{12} & I \end{bmatrix}$$

where $0 \in \mathbb{R}^{3 \times 3}$ is the null matrix of the appropriate size. By performing minor modification, the equations (9)–(10) can be rearranged using eq. (7) into the more favorable form

$$\dot{V}_{A1} = \xi_{A11}^A T_{A1} + \xi_{A12}^A T_{A2} + \xi_{A10}^A$$

$$\dot{V}_{A2} = \xi_{A21}^A T_{A1} + \xi_{A22}^A T_{A2} + \xi_{A20}^A$$

Equations (12), (13) provide a direct connection between the spatial velocity vectors of body $A$ and boundary impulsive forces. These equations are the final form of momentum conservation principle for general body $A$ with two handles, which are used in the further development procedure. As a final remark, it is worth noticing that $\xi_{12}^A = \left( \xi_{21}^A \right)^T$.

**Assembly phase** In this paragraph, the momentum conservation principle equations for a compound body $C$ will be constructed. Let us consider two bodies $A$ and $B$ interconnected by a general kinematic joint. For these bodies, as presented in the Figure 2, the momentum conservation equations can be written as

$$\dot{V}_{1} = \xi_{11}^A T_{1}^A + \xi_{12}^A T_{2}^A + \xi_{10}^A$$

$$\dot{V}_{2} = \xi_{21}^A T_{1}^A + \xi_{22}^A T_{2}^A + \xi_{20}^A$$

$$\dot{V}_{1} = \xi_{11}^B T_{1}^B + \xi_{12}^B T_{2}^B + \xi_{10}^B$$

$$\dot{V}_{2} = \xi_{21}^B T_{1}^B + \xi_{22}^B T_{2}^B + \xi_{20}^B$$
The goal of the assembly procedure is to construct the momentum conservation equations for the body $C$ in the analogous form

$$V_1^C = \xi_1^C T_1^C + \xi_{12}^C T_2^C + \xi_{10}^C$$  \hspace{1cm} (18)
$$V_2^C = \xi_{21}^C T_1^C + \xi_{22}^C T_2^C + \xi_{20}^C$$  \hspace{1cm} (19)

where handles $O_1$ and $O_2$ of the compound body $C$ are coincident with point $O_1$ for body $A$ and point $O_2$ for body $B$, respectively. The same form of the equations enables us to utilize divide and conquer paradigm to assembly the whole system into one super–body whose velocity depends on boundary impulsive forces. In other words, the expressions $\xi_{11}^C$, $\xi_{12}^C$, ..., $\xi_{20}^C$ for body $C$ have to be evaluated in terms of the known quantities for bodies $A$ and $B$. The procedure requires an intermediate calculation of the internal impulsive forces $T_1^A$ and $T_2^A$ using outboard forces $T_1^B$ and $T_2^B$. The presence of the interconnecting kinematic joint relates the bodies velocities as

$$V_1^B - V_1^A = Hq$$  \hspace{1cm} (20)

Matrix $H$ is the joint motion subspace that maps relative velocity $\dot{q}$ into the spatial velocity vector. To determine the interconnecting impulsive constraint force, equations (15)–(16) are substituted into (20), and then, projected onto the orthogonal complement of the joint motion subspace to give

$$D^T \left( \xi_{11}^B T_1^B + \xi_{12}^B T_2^B + \xi_{10}^B - \xi_{21}^A T_1^A - \xi_{22}^A T_2^A - \xi_{20}^A \right) = 0$$  \hspace{1cm} (21)

The impulsive constraint forces acting on the joint’s handle have to satisfy the Newton’s law that implies the following conditions

$$T_1^B = -T_2^A = D\sigma$$  \hspace{1cm} (22)

The combination of equations (21) and (22) enables us to acquire the definition of constraint impulsive Lagrange multipliers $\sigma$ as a function of outboard impulsive constraint forces $T_1^A$ and $T_2^B$

$$\sigma = -\left[ D^T \left( \xi_{11}^B + \xi_{22}^A \right) D \right]^{-1} D^T \left( \xi_{11}^B T_1^B + \xi_{21}^A T_1^A + \xi_{10}^B - \xi_{20}^A \right)$$  \hspace{1cm} (23)

The matrix inversions exist as both matrices $\xi_{11}^B$ and $\xi_{22}^A$ are symmetric and positive definite as long as the constraints imposed on the system are not dependent. Furthermore, let us define the following matrices

$$W = DCD^T = -D \left[ D^T \left( \xi_{11}^B + \xi_{22}^A \right) D \right]^{-1} D^T$$  \hspace{1cm} (24)
$$\beta = DCb = DCD^T \left( \xi_{10}^B - \xi_{20}^A \right)$$  \hspace{1cm} (25)

Hence, eq. (22) becomes

$$T_1^B = W\xi_{12}^B T_2^B - W\xi_{21}^A T_1^A + \beta = -T_2^A$$  \hspace{1cm} (26)

and describes how the impulsive forces at the outboard handles participate in the impulsive force at the joint interconnecting body $A$ and $B$. After the substitution of eq. (26) into eqs. (14) and (17), the momentum conservation equations for articulated body $C$ take the form

$$V_1^C = \left[ \xi_{11}^A + \xi_{12}^A W\xi_{21}^A \right] T_1^A + \left[ -\xi_{12}^A W\xi_{12}^B \right] T_2^B + \left[ \xi_{10}^A - \xi_{12}^A \beta \right]$$  \hspace{1cm} (27)
$$V_2^C = \left[ -\xi_{21}^A W\xi_{21}^A \right] T_1^A + \left[ \xi_{22}^A + \xi_{21}^A W\xi_{12}^B \right] T_2^B + \left[ \xi_{20}^A + \xi_{21}^A \beta \right]$$  \hspace{1cm} (28)

Because the appropriate handles of the compound body $C$ are coincident with the handles of body $A$ and $B$, some velocities and impulsive forces are equal

$$V_1^C = V_1^A \quad \text{and} \quad T_1^C = T_1^A$$  \hspace{1cm} (29)
$$V_2^C = V_2^B \quad \text{and} \quad T_2^C = T_2^B$$  \hspace{1cm} (30)
Figure 3: Assembly–disassembly process for multibody system

Equations (27) and (28) are equivalent to (18) and (19) if and only if
\[
\begin{align*}
\xi_{11}^C &= \xi_{11}^A + \xi_{12}^A W \xi_{21}^A, & \xi_{12}^C &= -\xi_{12}^A W \xi_{12}^B, & \xi_{10}^C &= \xi_{10}^A - \xi_{12}^A \beta, \\
\xi_{21}^C &= -\xi_{21}^B W \xi_{21}^B, & \xi_{22}^C &= \xi_{22}^B + \xi_{21}^B W \xi_{12}^B, & \xi_{20}^C &= \xi_{20}^B + \xi_{21}^B \beta,
\end{align*}
\] (31)

In this case the symmetry condition \( \xi_{12}^C = (\xi_{21}^C)^T \) is also preserved.

Equations (31) and (32) are the major formulas for the HDCA algorithm and let us assembly bodies \( A \) and \( B \) into a compound body \( C \). At this point it is possible to perform the hierarchic assembly of the multibody system based on the binary tree decomposition (see Figure 3). The efficient evaluation of the binary tree associated with the multibody system is a separate task and it will not be considered in this paper. The step finishes when the root node correlated to the mechanism’s base body connection is reached.

**Base body connection** Let us define the momentum conservation equations in a generic form for the whole system composed of one super–body as in equations (18)–(19). The mechanism has a closed loop topology and is connected to the fixed base by two kinematic joints at handle points \( O_1 \) and \( O_2 \). The impulsive constraint forces can be linked with impulsive Lagrange multipliers in the following manner
\[
T_1^C = D_1 \sigma_1 \quad \text{and} \quad -T_2^C = D_2 \sigma_2
\] (33)

whereas the joint velocities can be expressed as
\[
\dot{q}_1 = H_1^T V_1^C \quad \text{and} \quad -\dot{q}_2 = H_2^T V_2^C
\] (34)

Upon the substitution of equations (33) into the equations (18)–(19) projected onto the appropriate constraint subspaces we obtain
\[
\begin{bmatrix}
D_1 & 0 \\
0 & D_2
\end{bmatrix}
\begin{bmatrix}
\xi_{11}^C \\
\xi_{21}^C
\end{bmatrix}
\begin{bmatrix}
D_1 & 0 \\
0 & D_2
\end{bmatrix}
\begin{bmatrix}
\sigma_1 \\
-\sigma_2
\end{bmatrix}
+ \begin{bmatrix}
D_1 & 0 \\
0 & D_2
\end{bmatrix}
\begin{bmatrix}
\xi_{10}^C \\
\xi_{20}^C
\end{bmatrix}
= 0
\] (35)

The above equations enables us to compute unknown boundary impulsive Lagrange multipliers
\[
\begin{bmatrix}
\sigma_1 \\
-\sigma_2
\end{bmatrix}
= - \begin{bmatrix}
D_1 & 0 \\
0 & D_2
\end{bmatrix}
\begin{bmatrix}
\xi_{11}^C \\
\xi_{21}^C \\
\xi_{12}^C \\
\xi_{22}^C
\end{bmatrix}
\begin{bmatrix}
D_1 & 0 \\
0 & D_2
\end{bmatrix}
\begin{bmatrix}
\xi_{10}^C \\
\xi_{20}^C
\end{bmatrix}
\] (36)

Usually the matrix inversions in equation (36) exist. However, there are some circumstances in which the calculation of constraint impulsive loads is not so straightforward. The problems with matrix inversions may arise when the system passes near or through singular configuration or/and the system is redundantly constrained. Nevertheless, let us assume that the boundary impulsive
loads have successfully been evaluated at this phase. With boundary impulsive Lagrange multipliers known, the joint velocities connecting the system with fixed base can be computed as

$$\begin{bmatrix} \dot{q}_1 \\ -\dot{q}_2 \end{bmatrix} = H_1 0 H_2^T \begin{bmatrix} \xi_{11}^C & \xi_{12}^C \\ \xi_{21}^C & \xi_{22}^C \end{bmatrix} \begin{bmatrix} D_1 & 0 \\ 0 & D_2 \end{bmatrix} \begin{bmatrix} \sigma_1 \\ -\sigma_2 \end{bmatrix} + \begin{bmatrix} H_1 & 0 \\ 0 & H_2 \end{bmatrix}^T \begin{bmatrix} \xi_{10} \\ \xi_{20} \end{bmatrix}$$

(37)

Analogously, the above scheme can be easily adapted to cope with more handles as well.

**Disassembly phase**  As soon as the outboard impulsive constraint forces $T_1^C$ and $T_2^C$ are computed, they can be sent to the proceeding nodes of the binary tree and one can begin with the disassembly phase (cf. Fig. 3). This step allows the computation of all other impulsive constraint forces, absolute velocities and joint velocities in the system with the use of basic momentum conservation equations (eqs. (14)–(17)), handles coincidence conditions (eqs. (29)–(30)), and interconnecting impulsive constraint force relation (eq. (26)). Subsequently, projection of the equations (20) onto the joints’ motion subspaces result in the whole system’s joint velocities. The above phase is equivalent with the evaluation of the first set of Hamilton’s canonical equations (1). It allows to recursively calculate the derivatives of canonical coordinates recursively without the need of determination of the system’s Hamiltonian and its partial derivatives $\partial \mathcal{H} / \partial p$.

### 2.2 Derivatives of canonical momenta

In this section the formulas for time derivatives of joint canonical momenta are derived. For the sake of clarity, the starting point for further considerations are the equations of motion described with the use of absolute momenta.

**Equations of motion for constrained body**  Similarly to the derivations presented in the previous section, let us consider two bodies $A$ and $B$ again (see Fig. 2). Both could be a part of a longer chain, thus the pair have two handles each, which are connecting $A$ and $B$ together and associate them with the rest of the multibody system. The equations of motion for the physical or compound body $A$ and $B$ are expressed as follows:

$$\begin{align*}
\dot{P}_1^A + \dot{S}_1^A P_1^A &= Q_1^A + F_1^A + S_{12}^A F_2^A \\
\dot{P}_1^B + \dot{S}_1^B P_1^B &= Q_1^B + F_1^B + S_{12}^B F_2^B
\end{align*}$$

(38)

(39)

where subindex 1 refers to the first handle of body $A$, and subindex 2 is associated with the second handle of body $B$, respectively. Please note that the matrices $\dot{S}_1^A$ and $\dot{S}_1^B$ contain absolute, translational velocities associated with handle 1 of body $A$ and $B$, respectively. Expressions $F_1$ and $F_2$ are accordingly constraint reaction forces, which have to satisfy the Newton’s third law

$$F_1^A = -F_2^B$$

(40)

Now, by using eq. (40), let us substitute eq. (39) into eq. (38) to obtain

$$\begin{align*}
\dot{P}_1^A + S_{12}^A \dot{P}_1^B + \dot{S}_1^A P_1^A + \dot{S}_1^B P_1^B &= Q_1^A + S_{12}^A Q_1^B + F_1^A + S_{12}^A S_{12}^B F_2^B
\end{align*}$$

(41)

Equation (41) can be transformed into the more suitable form using the simplifying relation $\dot{S}_1^B = S_{12}^A + S_{12}^A$ to provide

$$\frac{d}{dt} (P_1^A + S_{12}^A \dot{P}_1^B) + S_1^A (P_1^A + S_{12}^A \dot{P}_1^B) = (Q_1^A + S_{12}^A Q_1^B) + F_1^A + S_{12}^A S_{12}^B F_2^B$$

(42)

By using the following substitutions and the conditions for coincidence of appropriate handles:

$$\begin{align*}
P_1^C &= P_1^A + S_{12}^A \dot{P}_1^B, \\
F_1^C &= F_1^A, \\
S_1^C &= S_1^A
\end{align*}$$

(43)

$$\begin{align*}
Q_1^C &= Q_1^A + S_{12}^A Q_1^B, \\
F_2^C &= F_2^B, \\
S_{12}^C &= S_{12}^A S_{12}^B
\end{align*}$$

(44)
equation (42) takes a more concise form

\[ \dot{P}_1^c + S_1^c P_1^c = Q_1^c + F_1^c + S_{12}^c F_2^c \]  \hspace{1cm} (45)

where \( P_1^c \) and \( Q_1^c \) are accumulated momenta and accumulated external loads. Please note that equation (45) describes the motion of the articulated body \( C \) that consists of physical or compound bodies \( A, B \) and the interconnecting joint. The form of equation (45) is similar to that presented in eqs. (38)–(39) for bodies \( A \) and \( B \). At this point it would be possible to perform the whole multibody system hierarchic assembly according to the binary tree decomposition. The purpose of the assembly phase is to obtain the equations of motion for the whole system that is treated algorithmically as a single body. The process is continued up to the moment at which the root node of the graph associated with the topology of the system is reached. The projection of the equation (45) onto the corresponding joint’s motion subspace results in the derivative of joint momenta

\[ \dot{p}_1 = H_1^T \left( Q_1^c - S_1^c P_1^c - S_{12}^c F_2^c \right) + H_1^T P_1^c \]  \hspace{1cm} (46)

The drawback associated with the equation (46) is the presence of unknown reaction force \( F_2^c \). Secondly, although equations (45) and (46) look promising, it would be more beneficial to express the equations of motion in terms of articulated momenta, which at this stage, are already available. The common procedure for constrained multibody systems is to artificially cut specific joint to produce open-loop system by introducing unknown constraint force. The idea formally leads to the extended Lagrangian and extended Hamiltonian functions for the considered multibody system. Derivation of which is presented in [16]. Let us assume that the mechanism \( C \) is cut at handle \( O_2 \), which coherently means that

\[ \overline{P}_2^c = T_2^c = -D_2 \sigma_2 \]  \hspace{1cm} (47)

The comparison of the accumulated momenta definition (43) with the articulated momentum conservation principle (compare (9)–(10)) gives

\[ P_1^c = \overline{P}_1^c + S_{12}^c \overline{P}_2^c \]  \hspace{1cm} (48)

\[ P_2^c = S_{21}^c \overline{P}_1^c + \overline{P}_2^c \]  \hspace{1cm} (49)

It can be deducted that there is a clear dependency between accumulated and articulated momenta. Substitution of equation (48) into eq. (45) and the use of eq. (47) results in

\[ \frac{d}{dt} \overline{P}_1^c + S_{11}^c \overline{P}_1^c + (S_{12}^c S_{12}^c + S_{12}^c S_{12}^c) \overline{P}_2^c - S_{12}^c D_2 \sigma_2 = Q_1^c + F_1^c \]  \hspace{1cm} (50)

The above relation simplifies to

\[ \frac{d}{dt} \overline{P}_1^c + S_{11}^c \overline{P}_1^c + S_{21}^c \overline{P}_2^c - S_{12}^c D_2 \sigma_2 = Q_1^c + F_1^c \]  \hspace{1cm} (51)

The projection of equation (51) onto the boundary joint’s motion subspace enables us to directly determine the derivate of the joint canonical momenta as the reaction force \( F_1^c \) is equal to zero

\[ \dot{p}_1 = H_1^T \left( Q_1^c - S_{11}^c \overline{P}_1^c - S_{21}^c \overline{P}_2^c + S_{12}^c D_2 \sigma_2 \right) + H_1^T \overline{P}_1^c \]  \hspace{1cm} (52)

in terms of the known articulated momenta. The above formalism can be easily extended to any joint encountered in the multibody system. The derivative of the joint’s canonical momenta \( k \) related to any body \( K \) in the system can be expressed in terms of already computed articulated momenta as

\[ \dot{p}_k = H_k^T \left( Q_k^c - S_{11}^k \overline{P}_1^c - S_{21}^k \overline{P}_k^c + S_{12}^k D_c \sigma_c \right) + H_k^T \overline{P}_1^c \]  \hspace{1cm} (53)
where subindex \(c\) indicates the cut joint, vectors \(\mathbf{P}_1^k\) and \(\mathbf{P}_c\) are articulated momenta for joint \(k\) and cut joint \(c\), respectively. The derivatives of shift matrices \(S_1^k\) and \(S_c\) contain translational velocities for handle \(k\) and \(c\) computed and measured with respect to the global, inertial reference frame. The application of this form of equations of motion enables us to include unknown reaction force at cut location \(c\) within the definition of the articulated momentum. Equation (53) can be used to calculate the time derivatives of joint momenta provided that accumulated external loads \(Q^K_1\) are known.

**Articulated external loads** To compute the derivatives of canonical momenta, it is still necessary to determine accumulated active loads vectors. Because of this fact, the analogous divide and conquer process can be carried out for the computation of necessary terms. Likewise before, let us define articulated external loads vector at handles locations \(Q_A^1\) and \(Q_A^2\), and associate them with external load vectors for a pair of bodies \(A\) and \(B\) at handle \(O_1\)

\[
Q_A^1 = \mathbf{Q}_1^A + S_{12}^A \mathbf{Q}_2^A \\
Q_A^2 = \mathbf{Q}_1^B + S_{12}^B \mathbf{Q}_2^B
\]  
(54)

As articulated loads \(\mathbf{Q}_1^A\) and \(\mathbf{Q}_1^B\) relate to the same kinematic pair connecting bodies \(A\) and \(B\), the following equality can be deducted

\[
\mathbf{Q}_2^B = -\mathbf{Q}_2^A
\]  
(56)

Substitution of equation (55) into (54) with the use of eq. (56) results in the dependency linking articulated loads with accumulated loads vectors

\[
\mathbf{Q}_1^A + S_{12}^A \mathbf{Q}_1^B = \mathbf{Q}_1^A + S_{12}^A S_{12}^B \mathbf{Q}_2^B
\]  
(57)

which can be rewritten using handles coincidence conditions for compound body \(C\) (compare with eqs. (29)–(30))

\[
\mathbf{Q}_1^C = \mathbf{Q}_1^A + S_{12}^A \mathbf{Q}_2^C
\]  
(58)

For the whole multibody system represented as a single entity, the articulated load vector \(\mathbf{Q}_2^C\) is zero as it refers to the cut joint. Due to this fact one can write the following relation

\[
\mathbf{Q}_1^C = \mathbf{Q}_1^A + S_{12}^A \mathbf{Q}_1^B = \mathbf{Q}_1^A = \mathbf{Q}_1^A
\]  
(59)

Please note that the accumulated load vectors transform in the same manner as accumulated momenta vectors presented in the previous subsection. Based on the divide and conquer algorithm, it is possible to develop an assembly–disassembly procedure, which exhibits logarithmic computational cost in parallel and computes all the necessary load vectors. If the external loads do not depend on the velocity terms, they can be computed during the joint velocity evaluation phase. Otherwise, external load vectors have to be expressed in terms of canonical momenta, or the procedure computing articulated loads have to be performed after the velocity evaluation phase is finished.

**Derivatives of canonical momenta** With all velocity terms, articulated momenta and external loads already known, the derivative of any canonical momenta \(k\) (related to body \(K\)) can be computed immediately as

\[
\dot{\mathbf{p}}_k = H_k^T \left( \mathbf{Q}_1^k - S_{12}^A \mathbf{P}_1^k - S_c \mathbf{P}_c + S_{ke} \mathbf{D}_e \sigma_c \right) + H_k^T \mathbf{P}_1^k
\]  
(60)

Equation (60) is equivalent to the second set of Hamilton’s canonical equations (2). It allows us to recursively calculate the derivatives of canonical momenta without the need of determination of the system’s Hamiltonian and its partial derivatives \(\partial \mathcal{H} / \partial \mathbf{q}\).
3 NUMERICAL TEST CASES

To verify the proposed HDCA algorithm, a closed–loop mechanism has been considered possessing \( N = 4 \) rigid bodies in the first case, and \( N = 8 \) bodies in the second case as presented in Fig. [ref]. The system has the total mass \( M = 4\text{kg} \) and characteristic length \( L = 2\text{m} \). The mechanism consists of \( N + 1 \) revolute joints and forms a parallel structure with the fixed base. Each body has mass \( m = M/N \), characteristic length of a body in the plane of motion \( l = L/(N - 2) \), and moment of inertia \( J_c = ml^2/12 \) with respect to the axis passing through the center of mass of a body. The initial position for both cases is illustrated in the Fig. 4, whereas initial joint velocities were set to zero. The mechanism should preserve the symmetry of the motion during the entire simulation. The results were competitively set against analogous Featherstone’s DCA algorithm [9] that solves forward dynamics problem by using acceleration based Newton–Euler’s equations. The non–stiff Dormand–Prince one step solver implemented in MATLAB (ode45) was used with the absolute and relative error tolerances equal to \( \delta = 10^{-6} \).

Figure 4: Numerical examples

![Numerical examples](figure4.png)

(a) Case N=4  
(b) Case N=8

Figure 5: Total energy preservation error

![Total energy preservation error](figure5.png)

(a) Case N=4  
(b) Case N=8

Figure 6: Constraints violation error on velocity level

![Constraints violation error on velocity level](figure6.png)

(a) Case N=4  
(b) Case N=8

Figure 5 presents the total energy preservation error as a function of the simulation time. It can be concluded from the plots that the HDCA algorithm is numerically more stable compared to the DCA and preserves the energy better for both cases. Figure 6 demonstrates the comparison of the constraint violation errors at the velocity level. As the HDCA algorithm combines the equations...
of motion with the velocity level algebraic constraints differentiated only once and DCA makes use of algebraic constraints at the acceleration level, there is huge discrepancy between the results in favor of Hamilton’s divide and conquer method. In case of HDCA, the constraint equations at the velocity level are satisfied within the numerical accuracy. This behavior has an impact on the constraints violation at the position level. The position constraint errors for DCA approach diverge quadratically in terms of the simulation time. On the other hand, the drift made by HDCA is negligible as indicated in Fig. 7. Figure 8 presents the position and the velocity of the middle joint for $N=4$ case and compares the HDCA and DCA algorithms results.

4 CONCLUSIONS

In this paper, a novel HDCA algorithm is presented for the simulation of constrained multi-rigid body system dynamics possessing closed-loop topologies. The HDCA approach utilizes Hamilton’s canonical equations of motion, which are recursively generated within the divide and conquer framework. The two-handle equations are formed for the spatial velocities of associated bodies and constraint impulsive loads at joints. Also the evaluation of the time derivatives of joint canonical momenta can be performed in a fully parallel manner. The comparison of the results with the analogous, acceleration based divide and conquer formulation indicates the predominance of the HDCA method. Sample closed-loop test cases demonstrate excellent constraint satisfaction at the position and velocity level as well as marginal energy drift. The properties of the method make the HDCA scheme worth considering in the applications to systems with more general topologies than those presented in the paper.

ACKNOWLEDGMENTS

This work has been supported by the National Science Centre under grant no. DEC-2012/07/B/ST8/03993.
REFERENCES


Simultaneous estimation of kinematic state and unknown input forces in rigid-link multibody systems

Ilaria Palomba, Dario Richiedei, Alberto Trevisani

Department of Management and Engineering (DTG)
Università degli Studi di Padova
Stradella S. Nicola 3, 36100 Vicenza, Italy
[ilaria.palomba, dario.richiedei, alberto.trevisani]@unipd.it

ABSTRACT

The knowledge of the external forces (and torques) applied to a multibody system is needed in several applications, such as monitoring or control. Unfortunately, such forces are often difficult to measure and therefore estimation techniques should be adopted. This paper proposes a novel two-stage approach making use nonlinear Kalman filters to estimate both unknown external forces and kinematic state variables in rigid-link multibody systems with negligible joint clearance. The approach consists in splitting the estimation process in two observers running simultaneously: a kinematic observer (first stage) and a force observer (second stage). The first one estimates the kinematic variables (i.e. position, velocity and acceleration) by just using position and acceleration measurements and the kinematic constraint equations, regardless of the knowledge of the external forces. The estimates obtained are then fed into the force observer, which estimates the external forces on the basis of the system dynamic model and the random walk approximation. Numerical assessment of the theory developed is provided through a slider-crank mechanism. The results achieved through the proposed approach are compared with those yielded by traditional unknown input observers based on single-stage dynamic observers, in order to show the advantages and the effectiveness of the new two-stage approach.

Keywords: force estimation, state estimation, Kalman filter, two-stage observer, rigid-link systems.

1 INTRODUCTION

State observers are widely used to estimate unmeasured state variables in multibody systems (MBS) whenever it is necessary to synthesize state feedback controllers, virtual sensors or fault detection schemes. An observer is usually designed to reconstruct missing state variables from accurate system models, and the knowledge of the time histories of the measured inputs and outputs. Clearly, model uncertainty may seriously compromise the accuracy of the estimates and affect their practical usefulness. The typical approach to state estimation in MBS relies on Kalman filters based on dynamic models [1, 2]. On the one hand, dynamic models involve a large number of parameters (e.g. geometrical dimensions and inertial properties of the links, friction models), which increase model uncertainty. On the other, the measured input of the observer should comprise all the forces (and torques) exciting the system analyzed, which are often difficult to be measured. This issue often prevents implementing state observers, unless the unknown forces are treated as unknown system parameters [3-5] and therefore included in the augmented state together with positions and velocities, and roughly modeled by means of the so called random walk. Random walk models are also used to estimate the unmeasured input forces in force observers, which can be useful in several applications, e.g. for evaluating the interaction forces with the environment in manufacturing or medical devices, or for synthesizing advanced control schemes. However, adopting such an approximate model, dictating that unknown forces should be represented through a constant value plus white noise (i.e. with infinite spectrum), may negatively affect the estimation of the kinematic state variables.

In the recent paper [6], it has been proved that the estimation of the kinematic variables in MBS, with rigid links and negligible joint clearance, can be based on just the kinematic constraint equations and on kinematic measurements, through the so-called Kinematic Kalman Filter
Indeed, the use of kinematic equations imposes a lower number of parameters compared to dynamic ones, and hence an higher accuracy, and does not require any force measurement. By exploiting such an approach, this paper proposes a novel technique for the simultaneous estimation of the kinematic state variables and the unknown force inputs for MBS with rigid links and negligible joint clearance. The basic idea consists in using acceleration measurements and splitting the estimation process into two observers running simultaneously: a KKF and a force observer. The KKF estimates positions, velocities and accelerations regardless of the knowledge of the external forces, and hence is unbiased by the uncertainty introduced by the unknown forces and by any force approximate model to be employed in the observer. The state estimates are then fed into the force observer as inputs. Such an observer estimates both the measured and the unmeasured forces, by employing simultaneously the inverse dynamic problem and the random walk, in a prediction-correction iteration aimed at compensating model uncertainty and measurement noise. In this sense the estimation process is said to be a two-stage one.

The paper is set out as follows: Section 2 briefly outlines the general scheme of nonlinear estimation and the Extended Kalman Filter (EKF) algorithm. The strategy proposed for the simultaneous estimation of the kinematic state and the unknown forces is described in Section 3. In Section 4 numerical tests proving the soundness of the proposed theory, are presented through on slider-crank mechanism. Finally, in Section 5 concluding remarks are given.

2 NON LINEAR STATE ESTIMATION

An observer aims at reconstructing the missing state of a system, \( \mathbf{x} \), on the basis of the knowledge of the differential equation representing the system model \( f_c \), the measurement model \( g \) and a set of measurements. The first model describes how the state propagates in time in the presence of external excitation due to inputs \( u \) and noise \( \delta \) or in the presence of not-null initial condition. The measurement model expresses the algebraic relation between the state and some noisy measurements \( y \). The general nonlinear model used by an observer is therefore represented through the first-order state-space model:

\[
\begin{cases}
\dot{x}(t) = f_c(x(t), u(t), \delta(t)) \\
y(t) = g(x(t), \delta^y(t))
\end{cases}
\]

In Equation (1) \( \delta^y \) is the vector of the measurement noise due, for instance, to sensor noise, bad calibration or misalignment.

Rather than employing Equation (1) directly in the synthesis of continuous-time filters, it is usually preferred the use of discrete-time filters based on difference equations. Hence, both the system equations and the measurement equations are modeled as discrete-time processes:

\[
\begin{cases}
x_k = f(x_{k-1}, u_{k-1}, \delta_{k-1}) \\
y_k = g(x_k, \delta^y_k)
\end{cases}
\]

having defined \( k \in \mathbb{N}^+ \) the time step index. The difference equation \( f \) is the discrete time state equation, which depends on both its continuous expression \( f_c \) and on the discretization scheme adopted.

In the MB field, the most widely used observers are those based on the Kalman Filter (KF) and on its non-linear extensions, as for instance the Extended Kalman Filter (EKF) and the Unscented Kalman Filter (UKF). One of the basic assumptions of KFs is that noise terms \( \delta \) and \( \delta^y \) are uncorrelated, Gaussian and white with zero mean:

\[
E(\delta \delta^T) = 0, \quad \delta \sim N(0, Q), \quad \delta^y \sim N(0, R)
\]

where \( Q \) and \( R \) are covariance matrices of, respectively, the model and the measurement noises. This type of filters are based on two main phases: the prediction and the correction. In the prediction phase, the current state variables are
computed through the uncertain models $f$ and the noisy input measurements. Then, when the next observation measurements $y_k$ are available, the estimated state is corrected by means of the error between the measured and the estimated output variables $y$, weighed through the filter gain. Therefore, the final estimation includes both the prediction based on the system model, and the closed-loop correction based on measurements.

In the following subsection, the main steps of the discrete EKF are briefly recalled to provide a clearer comprehension of the main issues related to the state (and input) estimation. For a more detailed discussion the interest reader is referred to [7].

### 2.1 Discrete EKF

The discrete EKF is the more straightforward extension to nonlinear systems (modeled through discrete model) of the KF originally developed for linear systems [8]. The general recursive scheme of the EKF is sketched in the block diagram in Figure 1.

**Figure 1:** EKF recursive algorithm.

In the diagram depicted in Figure 1, and in the theory developed hereafter, the hat indicates the estimated variables, the superscript "-" indicates the model-based a-priori estimates, while the superscript "+" denotes the a-posteriori estimates obtained by updating (i.e. correcting) the a-priori estimates on the basis of the measurements.

The EKF directly employs the nonlinear system equations $f$ and the measurement equations $g$ to perform both the state $\hat{x}_k$ and the observation $\hat{y}_k$ predictions (a-priori estimations). In contrast, it replaces the nonlinear model with its Jacobian matrices $F_{k-1}$ (the state transition matrix) and $H_k$ (the observation matrix), computed about the estimated state trajectory, in order to propagate the covariance matrix $P_k$ of the estimated state. Finally, the prediction is corrected through the error of output estimation $(y_k - \hat{y}_k)$, also called innovation, weighed through the filter gain $K_k$. The resulting estimation $\hat{x}_k^+$ is therefore obtained as:

$$\hat{x}_k^+ = \hat{x}_k^- + K_k (y_k - \hat{y}_k)$$  \hspace{1cm} (3)
As a consequence, provided that the system is observable, the estimation is forced to correctly track the sensor measurements, and model uncertainty is compensated.

3 THE TWO-STAGE OBSERVER

3.1 Overview of the two-stage observer

Although the state observers based on the prediction-correction iterations previously discussed partially compensate model uncertainty through filter innovation, the availability of correct models is necessary to ensure precise and stable estimations, as well as fast convergence. Additionally, the uncertainty affecting just some model equations or measurements can severely threaten the correctness of the estimates of the whole state, in the case of fully coupled observers. To this purpose, the method proposed in this paper aims at reducing the effect of the uncertainty due to the presence of unknown external forces by splitting the estimation process into two estimations carried out by two observes running simultaneously and only partially coupled: a KKF and a force observer. The KKF estimates positions, velocities and accelerations (collected in vector $\mathbf{x}_k^{aug}$, see Section 3.2.2), regardless of the knowledge of the external forces, and hence is unbiased by the uncertainty introduced by any approximate force model. Its estimates are then fed into the force observer as the inputs. In the force observer the equations describing the dynamic model of the system are implemented. The force observer estimates the unmeasured forces by also assuming random walk models to ensure a correct number of model equations, which should equal the number of forces applied to the system. The overall structure of the observer is summarized in Figure 2. It clearly shows the type of coupling between the two observers: the output of the KKF play the role of the measured input of the force observer. In contrast, the KKF does not depend on the force observer. This feature ensures that the usually major uncertainty introduced by the random-walk force model does not affect the estimation of the kinematic state.

This two-stage architecture of the observer allows exploiting all the benefits of kinematic estimation [6]. In the following subsections the kinematic and force observers are described in detail.

![Figure 2: Schematic representation of the proposed two-stage approach](image)

3.2 Kinematic observer

This Section briefly reviews the fundamental steps for synthesizing a kinematic observer. The interested reader should refer to [6] for a more detailed discussion.
3.2.1 First-order model formulation

The basic idea behind kinematic estimation is to exploit the kinematic constraint equations and a proper set of acceleration measurements to estimate other unmeasured kinematic variables. To this end, the constraint equations should be formulated as first-order Ordinary Differential Equations (ODEs) to fit the model of Equation (1), by properly defining the state vector \( \mathbf{x} \), the system inputs \( \mathbf{u} \), and the measured outputs \( \mathbf{y} \). The state of a MBS model for kinematic estimation is defined as a \( 2n \)-dimensional vector including the set of \( n \) independent coordinates (\( n \) is the number of degrees of freedom, dofs) \( \mathbf{z} \) and their first derivatives \( \dot{\mathbf{z}} \):

\[
\mathbf{x} = \begin{bmatrix} \mathbf{z}^T & \dot{\mathbf{z}}^T \end{bmatrix}^T
\]

(4)

With regard to the model input vector \( \mathbf{u} \), it should include at least \( n \) non-redundant acceleration measurements, denoted \( \mathbf{h} \). It is worth noticing that, in kinematic estimation, the input variables play the same role as the forces (or torques) in the traditional synthesis of state observers based on dynamic models, where the input vector includes the external forces.

In accordance with the definition provided for the state and the inputs and by taking advantage of the acceleration constraint equations, \( \dot{\mathbf{h}} = \mathbf{S}(\mathbf{z})\ddot{\mathbf{z}} + \dot{\mathbf{S}}(\mathbf{z}, \dot{\mathbf{z}})\mathbf{z} \), it is possible to define a first-order ODE-based kinematic model for MBSs:

\[
\dot{\mathbf{z}} = \begin{bmatrix} \dot{\mathbf{z}}_x \\ \dot{\mathbf{z}}_z \end{bmatrix} = \begin{bmatrix} \dot{\mathbf{z}}_x \\ \dot{\mathbf{z}}_z \end{bmatrix} = \begin{bmatrix} \mathbf{S}^T(\mathbf{z})\mathbf{S}(\mathbf{z})^{-1} & \mathbf{S}^T(\mathbf{z}) \end{bmatrix} \begin{bmatrix} \dot{\mathbf{z}}_x \\ \dot{\mathbf{z}}_z \end{bmatrix} - \mathbf{S}(\mathbf{z}, \dot{\mathbf{z}})\mathbf{z}
\]

(5)

In Equation (5) matrices \( \mathbf{S} \) and \( \dot{\mathbf{S}} \) are respectively the sensitivity coefficient matrix, \( \dot{\mathbf{h}} = \mathbf{S}(\mathbf{z})\dot{\mathbf{z}} \), and its time derivative. The existence of such a model depends on the choice of the system inputs, since it sets the existence of matrix \([\mathbf{S}^T(\mathbf{z})\mathbf{S}(\mathbf{z})]^{-1}\).

As far as the output vector \( \mathbf{y} \) it is concerned, it should include variables ensuring system observability. Generally speaking a system, whose model is known, is observable if its state is uniquely determinable from its measured inputs and outputs, and from the initial conditions [3]. In the kinematic estimation of MBSs a set of sensors ensuring adequate observability, even in the presence of uncertain state initial conditions, and noise, should include as many non-redundant position measurements as the number of dofs, since such measurements can capture the zero frequency dynamics.

3.2.2 Discrete time representation of the filter ODEs

The first-order ODE representation of the kinematic constraint equations stated in Equation (5), henceforth referred to as “the kinematic model”, should be transformed in a discrete time process. A general representation of the kinematic model in discrete time takes the following form:

\[
\mathbf{x}_k = \begin{bmatrix} \mathbf{z}_k \\ \dot{\mathbf{z}}_k \end{bmatrix} = f(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) = \mathbf{x}_{k-1} + \Gamma(\Delta t, \mathbf{f}_c(x(t), \mathbf{u}(t))) \Delta t
\]

(6)

where \( \Delta t \) is the discretization time, and \( \Gamma \) denotes the transformation made by the specific discretization scheme adopted. Equation (6) clearly shows that the discrete time filter allows directly estimating the positions \( \mathbf{z}_k \) and velocities \( \dot{\mathbf{z}}_k \) of the independent coordinates. In contrast, the estimation of the accelerations, which is required by the second stage of the proposed observer (the dynamic filter), requires a slight modification of the scheme. Indeed, in order to perform accurate discrete time closed-loop acceleration estimations with the prediction-correction iterations, the equations representing the accelerations should be included in the kinematic model, so that the correction \( \mathbf{K}_k(\mathbf{y}_k - \dot{\mathbf{y}}_k) \) is applied to compensate for measurement noise and model uncertainty. Indeed, the correction behaves as an optimal nonlinear filter whose
gain, and therefore bandwidth, are updated at each time step depending on the estimation error and on the noise, in accordance with the theory discussed in Section 2.

To this end the model defined in Equation (6) is augmented to include acceleration equations. In particular, by following the approach proposed in [6], acceleration equations are modeled by exploiting simultaneously both numerical derivation relations and kinematic constraint equations:

\[
\ddot{z}_k = \frac{\dot{z}_k - \dot{z}_{k-1}}{\gamma \Delta t} + \frac{\gamma^{-1}}{\gamma} \left[ S^T(z_{k-1})S(z_{k-1}) \right]^{-1} S^T(z_{k-1}) \left\{ u_{k-1} - \dot{S}(z_{k-1}, \dot{z}_{k-1}) \dot{z}_{k-1} \right\}
\]

where \( \gamma \) is a positive parameter defining the numerical derivation method adopted, in accordance with the Newmark’s first order interpolation.

Finally the augmented discrete time kinematic model is obtained including Equation (7) in Equation (6):

\[
x_{k}^{aug} = \left\{ \begin{array}{c}
x_k \\
\dot{z}_k
\end{array} \right\} = \left\{ \begin{array}{c}
f(x_{k-1}, u_{k-1}) \\
\dot{z}_k - \dot{z}_{k-1} + \frac{\gamma^{-1}}{\gamma} \left[ S_{k-1}^T S_{k-1} \right]^{-1} S_{k-1}^T \left\{ u_{k-1} - \dot{S}(z_{k-1}, \dot{z}_{k-1}) \dot{z}_{k-1} \right\}
\end{array} \right\}
\]

All the equations are supposed to be affected by noise.

### 3.3 Force observer

The synthesis of the second-stage observer relies on the dynamic model, to be formulated through a non redundant set of coordinates. Let us define the force vector \( F \), including both the known and unknown external forces acting on the MBS, which are collected respectively in vector \( F_1 \) and \( F_2 \):

\[
F = \left\{ F_1^T \ F_2^T \right\}^T
\]

In the case the dynamic equilibrium equations can be inverted by explicating the external forces as functions of the kinematic quantities \( z, \dot{z} \) and \( \ddot{z} \), and of the known forces, the following relation can be written:

\[
F_i = f_{12} (z, \dot{z}, \ddot{z}, F_2)
\]

where \( f_{12} \) is non linear functions representing the dynamic model. Once that the kinematic variables are known, Equation (10) represents a kinetostatic problem, and \( f_{12} \) is a set of algebraic equations with respect to the known state and the external forces. Therefore, they are not suitable to be directly employed in the observer model, which imposes the formulation of first-order ODEs. In contrast, they would be suitable to be employed in the so-called Disturbance Observers ([9]), which perform the estimation of the external forces without the prediction-correction “closed loop” iterations.

To make the dynamic model suitable for the observer synthesis, Equation (10) is transformed in a set of first-order difference equations (in the discrete time domain) and state, output and input vectors are defined. The state vector is the external force vector \( F \). The known forces \( F_i \) also set the measurement vector \( y_f \), \( y_f = F_i \), which allows computing the filter innovation (i.e. the correction). In order to ensure the system observability, \( y_f \) should include at least \( n \) non-redundant force measurements.

The first-order difference equations are formulated by relating the known forces at time \( k \) to the kinematic quantities and to the unknown forces computed at the previous step through the inverse dynamic equations. Then, in order to account for the approximation introduced by this
shift of a sample, an additive term $\delta_{k-1}$ representing noise is included in the model, to take advantage of the probabilistic nature of the state observer:

$$ F_{k} = f_{12}(z_{k-1}, \dot{z}_{k-1}, F_{k-1}) + \delta_{k-1} $$  \hspace{1cm} (11)$$

As a typical assumption, $\delta_{k-1}$ is modeled as white noise, even though band-limited (“colored”) noise could be easily adopted.

A second set of equations should be also included to model the unknown forces at time step $k$. Since no information about such forces is available, a random walk model is suggested. Basically, this approach approximates the dynamic of $F_{k}$ by assuming that the current value of $F_{k}$ is equal to the previous one, $F_{k-1}$, plus a noise terms $\delta_{k-1}$:

$$ F_{k} = F_{k-1} + \delta_{k-1}^* $$  \hspace{1cm} (12)$$

In the case no information about the spectrum of $F_{k}$ is available, $\delta_{k-1}^*$ is modeled as white noise, even though band-limited (“colored”) noise could be easily adopted whenever necessary.

The system model adopted in the second stage observer, for the estimation of the state vector $F$, is therefore represented through Equations (11) and (12), and the model input $u_{k-1}$ is the estimated kinematic augmented state, computed at the previous time step through the kinematic observer, $x_{k-1}^{aug} = (z_{k-1}^T, \dot{z}_{k-1}^T, \ddot{z}_{k-1}^T)^T$. The model assumed for estimation is therefore the following one:

$$ \begin{bmatrix} F_{k} \\ F_{k} \end{bmatrix} = \begin{bmatrix} f_{12}(F_{k-1}^{aug}, x_{k-1}^{aug}) \\ F_{k-1} \end{bmatrix} $$  \hspace{1cm} (13)$$

All the equations are supposed to be affected by noise.

As an alternative formulation, the dynamic model can be expressed by explicating $F_{2}$ as a function of the kinematic state and of the known forced $F_{1}$ (through function $f_{21}$ representing the dynamic model):

$$ F_{2} = f_{21}(z, \dot{z}, \ddot{z}, F_{1}) $$  \hspace{1cm} (14)$$

and the random walk model is assumed to approximate the dynamics of $F_{1}$:

$$ F_{k} = F_{k-1} + \delta_{k-1}^* $$  \hspace{1cm} (15)$$

In this case, the model assumed for estimation is the following one:

$$ \begin{bmatrix} F_{k} \\ F_{k} \end{bmatrix} = \begin{bmatrix} F_{k-1}^{aug} \\ f_{21}(F_{k-1}^{aug}, x_{k-1}^{aug}) \end{bmatrix} $$  \hspace{1cm} (16)$$

The selection of the most suitable formulation, between Equation (13) and Equation (16), is based on the existence of $f_{12}$ and $f_{21}$, which is in turn affected by the presence of singular configurations.

The analysis of the equations highlights that the number of unknown forces that can be estimated correctly (i.e. with bounded error) is not greater than the number of system degrees of freedom, provided that the same number of forces (e.g. the control forces exerted by the
actuators) is known (or at least accurately estimated indirectly, e.g. through current measurements). These conditions should not be regarded as a drawback of this method, since they also hold in the case of the disturbance observer and in the case of the estimation through a single-stage dynamic observer augmented with random walk models).

3.4 Method discussion

The method formulation proposed in the previous Sections highlights some of the benefits of the formulation here developed. Indeed it properly merges the use of kinematic models, dynamic models and the random walk approximation to reduce the uncertainty due to the presence of unknown forces.

On the one hand the first stage kinematic observer allows unbiased estimation of the kinematic augmented state, by preventing the drift due to the effect of unknown forces in the case of observers solely based on dynamic models. This is an advantage over traditional observers where simultaneous estimation of state and force is provided through dynamic MB models. Indeed, in the latter case estimation of the kinematic variables can be severely affected by the unknown forces and by the rough model of the random walk usually adopted, by also leading to unbounded estimation errors.

On the other hand, the proposed approach overcomes the “open-loop” nature of several disturbance observers recently adopted in modern motion control schemes, by employing the dynamic equations within the frame of a “closed-loop”, prediction-correction iteration to compensate for model uncertainty and sensor noise.

All these features justify the slight increase in the complexity of the proposed observer, which requires some more sensors to measure accelerations and impose performing kinematic estimation. However, as proved in [6] and as it is confirmed in Section 4, the low computational effort usually required by kinematic observers boosts the real-time implementability of the two-stage observer proposed.

4 NUMERICAL RESULTS

The theory developed has been applied to a single-dof, closed-chain, planar mechanism (slider-crank mechanism) lying on the vertical plane, whose scheme is depicted in Figure 3.

![Figure 3: Kinematic scheme of a slider-crank mechanism](image)

It has been assumed that the crank is driven by a known torque $C_c$, whose time history is shown in Figure 4.a. Additionally, the system is supposed to be equipped with an incremental encoder and a mono-axial accelerometer, measuring respectively the crank angular position $\theta_c$ and the
slider linear acceleration $a_s$. The simulated accelerometer and encoder signals are shown respectively in Figure 4.b and Figure 4.c. Measurement noise has been added on both the encoder and the accelerometer signals to reproduce real situations. In particular, Gaussian noises have been generated with amplitudes of, respectively, 0.003 rad (corresponding to an encoder resolution of 2000 pulses per revolution) for the encoder and 0.025 m/s$^2$ for the accelerometer.

The aim of the test is to estimate the crank angular velocity $\dot{\theta}_c$ and acceleration $\ddot{\theta}_c$, and the force applied to the slider. To this end, the two observers have been synthesized in accordance with the theory proposed:

- the kinematic observer estimating the augmented kinematic state $\mathbf{x}^{\text{aug}} = [\theta_c \ \dot{\theta}_c \ \ddot{\theta}_c]^T$, whose input is the slider acceleration $u = a_s$, and output the crank angular position $y = \theta_c$;
- the force observer estimating the external forces $\mathbf{F} = [C_c \ F_c]$, whose input is the augmented kinematic state $\mathbf{u}_f = \mathbf{x}^{\text{aug}} = [\hat{\theta}_c \ \hat{\dot{\theta}}_c \ \hat{\ddot{\theta}}_c]^T$, while the output is the torque driving the crank $y_f = C_c$.

Besides measurement noise, uncertainty on the model parameters has been assumed through some deviation between the nominal and actual values of some geometrical and inertial model parameters (see Table 1).
Table 1: Geometric and inertial parameters (nominal ± error)

<table>
<thead>
<tr>
<th></th>
<th>Crank</th>
<th>Rod</th>
<th>Slider</th>
</tr>
</thead>
<tbody>
<tr>
<td>Masses [kg]</td>
<td>0.75-0.013</td>
<td>1.56-0.016</td>
<td>1.2+0.012</td>
</tr>
<tr>
<td>Lengths [mm]</td>
<td>120-0.5</td>
<td>250+0.5</td>
<td>-</td>
</tr>
</tbody>
</table>

Figure 5: Estimated variables: crank angular velocity (a) and acceleration (b), force acting on the slider (c).

The angular velocity and acceleration of the crank estimated through the kinematic observer are plotted respectively in Figure 5.a and Figure 5.b, while the force exciting the slider, and estimated through the force observer, is shown in Figure 5.c. Just a short 2-second excerpt of the whole simulated test is shown. However the filter synthesized does not diverge and the estimation error remains bounded, as a prove of the stability (in the control theory sense) of the filter synthesized. The same figures also plot the estimations obtained with the traditional external force estimation approach, which estimates the kinematic states and the unknown force by means of a dynamic model augmented with a random walk to represent the external force $F_s$. The estimations with the traditional approach have been carried out using the same measurement set employing in the proposed observers. In particular, the torque $C_c$ driving the...
crank has been set as input of the traditional observer, while the crank angular position and the slider acceleration have been used to compute the innovation in the correction step. All the estimations have been carried out using the EKF algorithm, which is assumed as a representative example of filter. Nevertheless the theory is enough general to allow the use of other type of filters, such as the mentioned UKFs. Figure 5 also shows the actual (simulated) values of the variables to be estimated. The root mean square error (\(e_{\text{rms}}\)) between the estimated values and the corresponding actual ones is also provided in Figure 5 as a concise measure of the estimation accuracy. The results and the comparison with the actual state clearly highlight that better performances are ensured by the proposed approach, which always ensures smaller root-mean-square values of the estimation errors.

Finally, it is worth noticing the small computational effort required by the proposed approach, which is believed to ensure its real-time implementation. Indeed, with reference to the Matlab implementations developed, 0.54 seconds of CPU time were needed on average to run the 2 shown seconds in the case of the proposed approach (on a PC with Core i7-2700K, RAM 16 GB). A similar CPU time is also required by the benchmark observer, which took on average 0.59 seconds. Therefore, the proposed approach allows getting more accurate estimates of both state and unknown input forces, without compromising the real time implementation.

5 CONCLUSIONS

This paper has introduced a novel approach for the simultaneous estimation of the kinematic state of a MBS (i.e. position, velocity and acceleration), and of the external unknown forces exciting it. The basic idea consists in splitting the estimation into two stages by means of two observers running simultaneously and only partially coupled: a first-stage kinematic observer estimating the kinematic state, and a second-stage force observer. The latter receives the kinematic state variables estimated in the first one as the input. Conversely the kinematic observer is neither affected by the force observer nor by the external unknown forces, in order to reduce model uncertainties and get more accurate estimations.

The proposed approach allows fully taking full advantage of the considerable benefits of state (and state derivative) estimation based on kinematic constraint equations in MBS with rigid links and negligible joint clearance. Therefore, besides allowing more accurate estimations of the external forces, compared to traditional single-stage dynamic observers, the proposed method prevents the drift in state estimation due to the effect of unknown forces, and therefore ensures bounded errors in the estimates.

The comparison of the proposed method with a benchmark method, i.e. the traditional EKF based on the dynamic model and a force random walk model, corroborates all these theoretical considerations, by showing that more accurate estimates of both the state and the unknown input forces can be obtained through the two-stage approach. Additionally, since the kinematic observer is just based on simple geometrical relations, small computational efforts are required and therefore the real time implementation is straightforward.

REFERENCES


An analysis method for a system with mass and extremely flexible component and its application to analysis of deployable satellite

Yoshiki Sugawara*, Syuntaro Oshima*, Taku Chida*

*Faculty of Mechanical Engineering
Akita University
1-1 Tegata-Gakuen-machi, Akita-shi, Japan
ysugawara@mech.akita-u.ac.jp

ABSTRACT

Systems which consists of masses and extremely flexible components and are often employed for satellites to achieve various vast structures. Authors have proposed an efficient analysis method for such a system, which uses linear complementarity of the system. In this paper, the proposed method is applied for a spacecraft model which has extremely flexible components and numerical analysis is carried out for the system in order to characterize the behaviour of such system and demonstrate the applicability of the proposed method.

Keywords: Flexible Multibody System, Linear Complementarity Problem, Unilateral Contact

1 INTRODUCTION

In recent years, very large structures are employed for spacecraft in order to satisfy requirements of various missions. For example, S310-36 [1] project had demonstrated deployment experiment of a large antenna, IKAROS [2] had deployed large membrane in orbit for use as solar sail. As Figure 1 shows, the system of S310-36 project had triangle shape which consisted of net made of strings, three subsatellites and a main satellite. Three subsatellites were located on the vertexes of triangle and main satellite was located on the triangle’s centroid. The diameter of the string constituting the net was about 1[mm] and the length of one side of the triangle was more than 10[m]. The system was stowed in the rocket fairing which diameter was about 30[cm] and deployed in orbit successfully.

In order to configure very large structure for spacecraft, extremely flexible components are employed and they connect some masses, which correspond to some kinds of equipment for satellite system. In the following, we call such a system “SMEF: System with Masses and Extremely Flexible components”. In general, it is quite difficult to carry out experimental
validation by ground test before its launch because influences of gravity and drag force are significant for SMEF. Therefore, numerical analysis is usually performed as feasible validation method. However, dynamical behavior of SMEF includes various vibrations of very high and low frequencies, and additionally SMEF is subjected to large deformation. These features of the SMEF’s behavior lead to the degradation of computational performance. Consequently, unrealistically large amount of computational time is required for precise analysis worse when conventional analysis method is employed, and reliability of existing analysis methods applicable to SMEF are still insufficient. Therefore, it is difficult to analyze the SMEF’s behavior for emergency case or use the analytical results for control of them. Consequently, analysis method with moderate accuracy and small calculation cost is quite important and is strongly required.

There is a characteristic feature of SMEF’s behavior. When the flexible component has slack (hereinafter referred to as “slack state”), the influence of the deformation of flexible component on dynamics of whole system is small and negligible. On the other hand, when the flexible component has tensile force (hereinafter referred to as “no-slack state”), the influence of the deformation of flexible component on dynamics of whole system is significant. Therefore, it can be expected that the analysis can be executed without considering the effect of flexible components if the components is in slack state and that such a treatment would save large amount of calculation time. Then, it is required for numerical analysis of SMEF to determine if the state of the flexible component is in slack state or no-slack state when aforementioned treatment is carried out. However, as the number of components becomes large, such determination problem usually leads to combinational explosion and it follows that the required time for analysis become large.

According to above discussions, there are two kinds of state transitions in the behavior of flexible component of SMEF. One is the state transition from no-slack state to slack state (hereinafter referred to as “state transition I”). The other is the state transition from slack state to no-slack state (hereinafter referred to as “state transition II”). Authors have found the analogy between two state transitions in SMEF’s behavior and two of state transitions in simultaneous multiple contact problem. There are several analysis methods for the behavior of simultaneously multiple contact problem. Among these methods, a method proposed by Pfeiffer et al. [3] is focused on due to its calculation efficiency and correctness. They utilize the complementarity of the system and describe the state transition problems by linear complementarity problem (hereinafter referred to as “LCP”). Then, analogously, state transition problem in SMEF can be also described and solved by LCP, that is, the state of SMEF can be determined effectively. The effectiveness of the proposed method, i.e. the analysis method of SMEF’s behavior by LCP formulation, has already demonstrated by the use of simple models in previous study [4]. Therefore, the aim of this study is set to application of the proposed method to analysis of deployable satellite’s behavior and investigation of the applicability of the proposed method to other systems.

This paper is organized as follows. In Section 2, the proposed analysis method for SMEF’s behavior is overviewed briefly and several features are indicated. In section 3, deployable satellite with extremely flexible components is introduced as analyzed object and formulation of the system based on the proposed method is carried out. In section 4, numerical analyzes are performed by the use of results obtained in Section 3 in order to validate the proposed method, and furthermore the applicability of the proposed method is discussed in the point of calculation time. Finally, conclusions and future works are described in Section 5.

2 OVERVIEW OF PROPOSED METHOD

In this chapter, general SMEF is introduced as analyzed object in our study and its feature is defined. Furthermore, the proposed method is overviewed briefly by applying it to simple model which a kind of SMEF.
2.1 General SMEF to be analyzed

In order to overview the proposed method, analyzed object, i.e. general SMEF, is introduced and several features are defined in the following part. Figure 2 shows a general SMEF to be analyzed in this study.

![Figure 2 General SMEF to be analyzed in this study](image)

As Figure 2 shows, masses are connected by strings which are employed as flexible component. It is assumed that there is no branching of the strings and that mass is point mass, that is inertia moment is set to 0 and consequently rotational motions of masses are not considered in this study. For the sake of explanation of the proposed method, fundamental component as shown in right part of Figure X is focused on in the following part. The system shown in the right part of Figure X consists of mass \( i \), mass \( i+1 \) and flexible component \( k \) which connects both masses. Length of the flexible component \( k \) is \( l_k \), displacements in the direction \( x \) and \( y \) of mass \( i \) and mass \( i+1 \) are \( x_i, y_i, x_{i+1} \) and \( y_{i+1} \), respectively. Then, introducing the relative slack displacement \( s_k \) which is the difference between the length of the flexible component \( k \) and the distance from mass \( i \) to mass \( i+1 \), \( s_k \) is given by a function which consists of \( q_i = [x_i, y_i]^T \) and \( q_{i+1} = [x_{i+1}, y_{i+1}]^T \) as follows.

\[
s_k = f_k(q_i, q_{i+1})
\]  

(1)

Furthermore, \( \dot{s}_k \) and \( \ddot{s}_k \) are derived from Eq. (1) as

\[
\dot{s}_k = W^T_i(q)\dot{q}, \quad \ddot{s}_k = W^T_i(q)\ddot{q} + w_i(q, \dot{q}),
\]

(2)

where \( q = [q_i^T q_{i+1}^T \cdots q_N^T]^T \) and \( N \) is the number of masses in the system. Furthermore, \( W^T_i(q) \) and \( w_i(q, \dot{q}) \) are matrix and vector which are obtained when \( \dot{s}_k \) and \( \ddot{s}_k \) are described by forms shown in Eq. (2). Supposing \( \sigma_k \) is the magnitude of the tensile force in the flexible component \( k \), equation of motion for the whole system is given by

\[
M \ddot{q} - h - W_\sigma = 0,
\]

(3)

where \( M = \text{blockdiag}(M_1, M_2, \cdots, M_N), \quad M_i = \text{diag}(m_i, m_i) \) and \( m_i \) is mass of the mass \( i \). \( h \) is a vector which consists of external force applied on each mass. Moreover, \( W = [W_1, W_2, \cdots, W_P]^T \) and \( \sigma = [\sigma_1, \sigma_2, \cdots, \sigma_P]^T \), and \( P \) is the number of the flexible components. Note that Eq. (3) satisfies for \( \sigma \) which includes zero components, i.e. no tensile force, however, the zero component in \( \sigma \) and corresponding component in \( W \) can be excluded from \( \sigma \) and \( W \) respectively, for the sake of convenience in formulation by the proposed method. In the other word, structure of \( \sigma \) and \( W \) are variable depending on the state of flexible components and an example is given as follows for the case \( P = 3, \quad \sigma_1 \neq 0, \quad \sigma_2 = 0, \quad \sigma_3 \neq 0 \),

\[
\sigma = [\sigma_1, \sigma_2, \sigma_3]^T, \quad W = [W_1, W_2]^T
\]

(4)
2.2 State transition I

In order to consider the state transition I, the index of the flexible component which is in no-sack state is given by $k_r$. Then, according to Eq. (2), the relative slack acceleration is given by

$$\ddot{s}_{k_r} = W_{k_r}^T \ddot{q} + w_{k_r}.$$

Furthermore, consideration of all flexible components which is in no-sack state yields

$$\ddot{s} = \overline{W}^T \ddot{q} + \overline{w},$$

where $\ddot{s}$, $\overline{w}$ and $\overline{W}$ are vectors and matrix which consist of $\ddot{s}_{k_r}$, $w_{k_r}$ and $W_{k_r}$, respectively, and components in each vector and matrix are stacked in ascending order of index $k_r$. Then, introducing a vector $\sigma$ which components are $\sigma_{k_r}$ and stacked in ascending order of index $k_r$ and considering Eq. (3) and the complementarity between $\ddot{s}_{k_r}$ and $\sigma_{k_r}$, state transition I is described by

$$\ddot{s} = A \sigma + B,$$

$$\ddot{s}_{k_r} \geq 0, \quad \sigma_{k_r} \geq 0, \quad s_{k_r}, \sigma_{k_r} = 0,$$

where $A$ and $B$ are matrix and vector which consist of system parameter, i.e. $M$, $\overline{W}$, $h$ and $\overline{w}$. Note that Eq. (8) and (9) form a LCP.

2.3 State transition II

In the same fashion as Pfeiffer’s method for contact problem, state transition II is divided into two phases as shown in Figure 3. One is defined as “Extension phase” and the other is defined as “Shrink phase”. At first, system is in no-sack state ([$A$]) and $s_k < 0$. Once $s_k$ becomes zero ($[B]$), impulsive tensile force begins to be applied to the string, and then the system enters the Extension phase ($[B] \rightarrow [C]$) where the strings is subjected to infinitesimal extension during infinitesimal time. In the Extension phase, impulse generated by impulsive force makes $s_k$ zero and is stored in the flexible component. At the moment that $s_k$ becomes zero, direction of the impulse becomes opposite and magnitude of the impulse becomes less than or equal to the original value, that is to say that exchange of impulse occurs. Finally, system enters Shrink phase ($[C] \rightarrow [D]$) and the exchanged impulse produces $s_k$ of positive value in the phase.

In order to describe the above mentioned state transition II, Extension phase is focused on at first. Then, integrating Eq. (3) in Extension phase yields

$$\ddot{s}_{k_r} = \int \ddot{s}_{k_r} dt = \int \ddot{s}_{k_r} (t) dt = \ddot{s}_{k_r} (t),$$

$$s_{k_r} = s_{k_r} (t).$$

Figure 3 Assumption on the State transition II
where \( \dot{q}_E = \dot{q}(t_E) \), \( \dot{q}_L = \dot{q}(t_L) \),

\[
\mathbf{lim}_{t_E \to t_L} \int_{t_L}^{t_E} \dot{\mathbf{\Sigma}} dt = \mathbf{\Sigma}_E.
\]

and \( \dot{\mathbf{\Sigma}} \) is a vector which components are \( \sigma_{k_i} \), where \( k_i \) is the index for the strings which slack displacement is zero, and \( \sigma_{k_i} \) are stacked into \( \dot{\mathbf{\Sigma}} \) in ascending order of index \( k_i \). Furthermore, components of \( \dot{W} \) are \( W_{k_i} \) which are stacked in ascending order of index \( k_i \). Because the magnitude of \( h \) in Eq. (3) is quite small compared to \( \mathbf{\Sigma} \), it is assumed that the integration of \( h \) is negligible. Supposing that \( \dot{\mathbf{\Sigma}}_{k_i} = \dot{\mathbf{\Sigma}}_{k_i}(t_E) \) and \( \dot{\mathbf{\Sigma}}_{k_i} = \dot{\mathbf{\Sigma}}_{k_i}(t_L) \) are velocities of relative slack displacements for strings corresponding to \( \sigma_{k_i} \) at \( t = t_L \) and \( t = t_E \) and introducing two vectors \( \dot{\mathbf{\Sigma}}_E \) and \( \dot{\mathbf{\Sigma}}_L \) which consist of \( \dot{\mathbf{\Sigma}}_{k_i} \) and \( \dot{\mathbf{\Sigma}}_{k_i} \) stacked in ascending order of index \( k_i \), following expression is obtained form Eq. (2) as

\[
\dot{\mathbf{\Sigma}}_E = \dot{W}^T (\dot{q}_E - \dot{q}_L) + \dot{\mathbf{\Sigma}}_L.
\]

Next, Shrink phase is focused on. In a similar fashion of deriving Eq. (12) and (13), LCP for Shrink phase is derived as follow,

\[
\dot{\mathbf{\Sigma}}_S = \dot{A}_S \mathbf{\Sigma}_S + \dot{B}_S,
\]

where \( \dot{A}_S \) and \( \dot{B}_S \) consist of system parameters, i.e. \( M \), \( \dot{W} \), \( \dot{\mathbf{\Sigma}}_E \).

Note that \( \dot{\mathbf{\Sigma}}_{k_i} \) is introduced in order to prevent rupture of strings due to influence of simultaneous multiple impulsive tensional force and it corresponds to the treatment in the analysis method proposed by Pfeiffer et al. [3]. Note that the detail discussion about introduction of \( \dot{\mathbf{\Sigma}}_{k_i} \) is omitted in this paper due to clarity of explanation.

Summarizing above discussion on the state transition II, solutions of two LCPs, i.e. Eq. (12), (13), (14) and (15), is obtained from the relative velocities at the moment right before the state transition II, and the solutions yield relative velocities after state transition II and impulses associated with that transition. Furthermore, generalized velocities associated with the state transition II is derived from relative slack velocities and impulses obtained from above LCPs as

\[
\dot{q}_S = M^{-1} \dot{W} (\Sigma_p + E \mathbf{\Sigma}_E + \mathbf{\Sigma}_S) + \dot{q}_L,
\]

where \( \dot{q}_L = \dot{q}(t_L) \), \( \dot{q}_S = \dot{q}(t_S) \).
2.4 Studies on features of the proposed method by an application to simple model and comparative study

In this section, behavior of simple SMEF is analyzed by the proposed method in order to study the features of the proposed method. Furthermore, obtained result is also compared with a result by FEM which is one of the general conventional analysis methods for flexible element. The left part of Figure 4 shows the simple model employed in this section and the right part of Figure 4 is FEM model used for comparative study. The simple model consists of one string and one mass. One end of the string is attached to the mass and the other end is attached to ceiling. Each parameter of the system is shown in Figure 4. Note that the specifications on material of the string should be given in FEM analysis and then the string is assumed to be thin aluminium wire as shown in Figure 4 in the FEM model. On the other hand, in the proposed method, the coefficient of impulse exchange should be given and it is set to 1 for comparative study because FEM model does not have dissipative element in the model. At initial state, the string has slack and the distance from the ceiling to the mass is 0.57[m]. In the analysis by the proposed method, one dimensional movement is focused on, i.e. vertical motion of the mass. FEM model is formulated by the use of L2-T1 model, which is kind of nonlinear FEM called as Absolute Nodal Coordinate Formulation (ANCF) [5]. FEM model uses 4 elements for the string and one element for the tip mass as right part of Figure X shows.

The results of both analyzes are given in Figure 5. As Figure 5 shows, behaviours of both analyzes are almost similar to each other. However, differences are observed at about 0.3[s] and 0.9[s]. These differences come from the assumption that string is made of elastic material i.e. aluminium, and such elasticity of the string leads to elastic deformation at 0.3[s] and 0.9[s] at which impulsive tensile force is applied to the string. Furthermore, small horizontal motion is induced in FEM model due to the influence of the string’s transverse stiffness while completely vertical motion is achieved in the proposed method. Because the expected feature of the proposed method is moderate accuracy and fast calculation time, the calculation times of 1[s] analyzes are compared between two methods and the results are shown in Table 1. As Table 1 indicates, the calculation time of the proposed method is more than 99% shorter than that of conventional method and it can be confirmed that the proposed method achieve the expected feature. In the conventional method, initial state determination problem has to be solved in order to prevent the unrealistic excessive stress and such a treatment requires additional preliminary analysis. In the case of analysis shown in Figure 4, it takes 1932[s] to determine the initial state by the preliminary analysis. On the other hand, the proposed method does not have such a treatment and the propose method is also advantageous in that point.

![Figure 4 Simple model for study of features of the proposed method](image-url)
3 APPLICATION TO DEPLOYABLE SATELLITE

A system as deployable satellite is introduced as Figure 6 and it has similar structure with the system of S310-36 project [1]. The proposed method is applied to the system and mathematical expression is derived in this chapter. The introduced system has equilateral-triangular shape in the fully deployed state and consists of four masses connected each other by several strings. A mass is located on the centroid of the triangle and each of three masses is located on each vertex of the triangle. It is assumed that deployment of the system is taken place within a plane, thefore mathematical expression is derived for two dimensional behavior of the system. Furtermore, following assumptions are introduced for the system:

- All masses are point mass.
- Weight of strings are quite light and negligible.
- Bending stiffness of the strings quite small and negligible.
Figure 7 Fundamental element which composes the whole system

Mass of the point mass on the centroid is $m_4$, and masses of the point masses on the vertexes are $m_1$, $m_2$ and $m_3$, respectively, and $m_1$, $m_2$, $m_3$ and $m_4$ are used as names of corresponding point masses in what follows for convenience. Then, equation of motion is given by

$$Mq - W\Sigma = 0,$$

where

$$q = [x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4]^T,$$  

$$W = W^T \ddot{q},$$

$$M = \text{blockdiag}(M_1, M_2, M_3, M_4),$$

$$M_i = \text{diag}(m_i, m_i),$$

$$\Sigma = [\sigma_{i1}, \sigma_{i2}, \sigma_{i3}, \sigma_{i4}, \sigma_{i5}, \sigma_{i6}, \sigma_{i7}, \sigma_{i8}]^T,$$

$$s = [s_{i1}, s_{i2}, s_{i3}, s_{i4}, s_{i5}, s_{i6}, s_{i7}, s_{i8}]^T,$$

$x_i$ and $y_i$ ($i = 1, 2, 3, 4$) are generalized coordinates of $m_i$, $s_{ij}$ is relative slack displacement of string between $m_i$ and $m_j$, $\sigma_{ij}$ is tensile force of string between $m_i$ and $m_j$. Structure of $W$ and $\Sigma$ given by Eq. (21) and (24) depends on the value of $\sigma_{ij}$ which is the component of $\Sigma$, and $\Sigma$ and $W$ are composed by removing $\sigma_{ij}$ of zero value and column vector of $W$ corresponding to $\sigma_{ij}$ of zero value. The system of Figure 6 consists of fundamental elements shown in Figure 7, which illustrates the relations of $s_{ij}$ and $\sigma_{ij}$ schematically. In the figure, $L_{ij}$ means length of the string between $m_i$ and $m_j$. It is clear that the fundamental element has the same structure with the system used in previous chapter for the explanation of the proposed method and that the proposed method can be applied easily without any special treatment, hence detail procedure of application of the proposed method is omitted in order to make point of the paper clear-cut.

4 NUMERICAL ANALYSIS FOR DEPLOYABLE SATELLITE BY PROPOSED METHOD

In this chapter, numerical analyzes of the deployable satellite’s behaviour by the use of the proposed method are carried out. In the numerical analyzes, deployment by centrifugal force is demonstrated and influences of the several parameters on the deployment behaviour are studied.

4.1 Conditions for numerical analyzes

The length of all of the strings which connect $m_4$ with $m_1$, $m_2$ and $m_3$ is 8.8 [m] and the length of other strings is determined automatically so that the deployed shape of satellite becomes equilateral triangle. Coefficient of impulse exchange for all of the strings are set to 0.2. In order to characterize the initial state, three parameters $L_{i4}$, $L_{24}$ and $L_{34}$ are introduced as Figure 8.
shows and they represents the distance from $m_4$ to $m_1$, $m_2$, and $m_3$, respectively. For simplicity, all of the neighboring direction from $m_4$ to $m_1$, $m_2$, and $m_3$ has angle of $2\pi/3$[rad.] at initial state. It is supposed that the system has rotational velocity $\omega_{ini} = \pi/3$[rad.] at initial state for deployment. In order to study the dependence of the deployable satellite’s behaviour on parameters, influences of mass and initial state are investigated and three cases of numerical analysis are performed for three sets of parameters shown in Table 2. In case 1, all point mass on the vertexes of the triangle has same mass and all of $L_{14}$, $L_{24}$, and $L_{34}$, has same value. In this study, Case 1 is defined as nominal case. On the other hand, $m_i$ has different value from those of $m_2$ and $m_3$ in Case 2. In Case 3, the distance $L_{34}$ is different from $L_{14}$ and $L_{24}$.

Figure 8 Initial state of the system

Table 2 Parameter sets for three cases

<table>
<thead>
<tr>
<th></th>
<th>Case 1</th>
<th>Case 2</th>
<th>Case 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>$m_1$ [kg]</td>
<td>5.0</td>
<td>6.0</td>
<td>5.0</td>
</tr>
<tr>
<td>$m_2$ [kg]</td>
<td>5.0</td>
<td>5.0</td>
<td>5.0</td>
</tr>
<tr>
<td>$m_3$ [kg]</td>
<td>5.0</td>
<td>5.0</td>
<td>5.0</td>
</tr>
<tr>
<td>$m_4$ [kg]</td>
<td>35.0</td>
<td>35.0</td>
<td>35.0</td>
</tr>
<tr>
<td>$L_{14}$ [m]</td>
<td>7.0</td>
<td>7.0</td>
<td>7.0</td>
</tr>
<tr>
<td>$L_{24}$ [m]</td>
<td>7.0</td>
<td>7.0</td>
<td>7.0</td>
</tr>
<tr>
<td>$L_{34}$ [m]</td>
<td>7.0</td>
<td>7.0</td>
<td>5.6</td>
</tr>
</tbody>
</table>

4.2 Result of numerical analysis

Numerical analyzes are performed for the aforementioned three cases, and 20 [s] time histories of response are derived for every point mass. Then, trajectories of each point mass are calculated by use of the obtained time histories and plotted in Figure 9 to 11. Figure 9, 10 and 11 are trajectories of Case 1, 2 and 3, respectively. Each figure has 4 small figures and they are trajectories of every 5[s]. Note that the trajectories of all strings are not depicted in those figures because the time histories of string’s behavior is not calculated due to feature of the proposed method. In each figure, red solid line, green dashed line, blue dashed-dotted line and black solid line indicate the trajectories of $m_1$, $m_2$, $m_3$ and $m_4$, respectively, and horizontal axis and vertical axis are $x$-axis and $y$-axis, respectively. In order to show the directions of movement, arrows are depicted in each figure. Note that $m_4$ does not move in Case 1, hence trajectory of $m_4$ does not appear in Figure 9.

In Case 1 as shown in Figure 9, all point masses on vertex have same mass and initial configuration is centrally symmetric. Hence, movement of masses are also centrally symmetric and the trajectory converges to exact circle within 5[s]. Here, convergence of the trajectory to
exact circle means that the satellite achieve its full deployment. On the other hand, \( m_1 \) is 20% larger than \( m_2 \) and \( m_3 \) in case 2. Note that Case 2 corresponds to the case that amount of fuel for propulsion in a subsatellite is different from others and it is worth to consider such a case because it may occur in practical system. It is clear that whole system begins to move left after the all strings connected to \( m_4 \) reach their natural length. It is guessed that the differences of mass of \( m_1, m_2 \) and \( m_3 \) lead to the asymmetry of momenta in the system and consequently such an asymmetry causes the translational motion of the whole system. Because of such an asymmetry property, it takes a longer time than case 1 for trajectory to converge to exact circle and such results may have significant influence on the efficiency of the operation in actual use.

In Case 3, \( m_1 \) is closer to \( m_4 \) than \( m_2 \) and \( m_3 \) in the initial state. This corresponds to the case that there is a time lag between the moment of release of subsatellite from the folded state which corresponds to the configuration in rocket fairing, and it is assumed that \( m_2 \) and \( m_3 \) are released earlier than \( m_1 \) in Case 2. In space systems, there is often time lag problems due to unexpected disturbances and failure caused by special space environment, therefor it is also important to consider the condition like Case 3. As the trajectory of 0 to 5[s] in Figure 11 shows, the string between \( m_1 \) and \( m_4 \) reaches to its natural length later than other two strings, i.e. strings between \( m_2 \) and \( m_4 \), and \( m_3 \) and \( m_4 \). Consequently, such a time lag causes an asymmetry in deployment process and it takes longer time in Case 3 for the system to converge to exact circle than other cases.

Advantage of the proposed method is low calculation cost and hence fast calculation is expected. Required times for 20[s]-analyzes of Case 2 and 3 were both less than 50[s] by the use of 1.2GHz clock CPU. On the other hand, it took about 200[s] to analyze 20[s] response of Case 1. Even in case1 which needed the longest calculation time among three cases, the proposed method require 10 times longer time than the desired time of numerical analysis. Note that behavior of Case 1 converges to steady state more quickly than other cases and steady state case requires more determination process of state transition because every strings has its natural length in steady state and string of natural length have two possibilities, that is, to keep its length and to begin to slack. In practical use, steady state case is trivial. Therefore, longer time required for analysis of Case 1 is not matter essentially. Accordingly, it can be said that the proposed method demonstrated efficient calculation.

![Figure 9 Result of numerical analysis for Case 1](image-url)
Figure 10 Result of numerical analysis for Case 2

Figure 11 Result of numerical analysis for Case 3
5 CONCLUSIONS

In our study, an effective analysis method is proposed in order to analyze the behavior of SMEF and its effectiveness is demonstrated by the application to simple model. Furthermore, in order to apply the proposed method to deployable satellite, mathematical expression for the deployable satellite is derived and consequently its behavior is numerically analyzed by the use of the proposed method. In the numerical analyzes, three cases of parameters are used and feature of behavior in each case is studied.

The proposed method did not deal with rotational motion of masses, i.e. body motion, therefore it is required for the method to be developed so that behavior associated with rotational motion can be analyzed. Furthermore, the method has to be applied to more practical application case and experimental validation for such a case also has to be performed.

ACKNOWLEDGMENT

This work was supported by KAKENHI 26420165.

REFERENCES


Approximative modeling of compliant mechanism dynamics

M. Roesner*, R. Lammering*, R. Friedrich

* Helmut-Schmidt-University / University of the Federal Armed Forces
Institute of Mechanics
Holstenhofweg 85, 22043 Hamburg, Germany
malte.rosner@hsu-hh.de

ABSTRACT
In this work, a novel approach towards geometrically nonlinear modeling of compliant mechanisms is presented which serves as an origin for subsequent model order reduction procedures to provide linear, efficient and accurate models for control purposes. As 3D finite element modeling of compliant mechanisms results in very large-scale systems, both model reduction and real-time controlling of the mechanism are hard to achieve. This and the fact that in compliant mechanisms mostly the flexure hinges contribute to the overall mobility motivates a procedure which is based on partitioning the structure into elastically deformable hinges and stiff linkages. Model order reduction by component mode synthesis and Krylov subspaces enables the significant reduction of the number of degrees of freedom. Unlike common modeling techniques, the reproduction of the non-negligible nonlinear behavior is assured and contributes to precise approximation as well as the fact that not only the flexure itself deforms but also adjacent structures. Furthermore, the proposed methodology is applicable to all kinds of flexure hinges with concentrated compliances as well as compliant mechanisms of complex geometric shape including spatial loading cases. The application of the proposed method to an exemplary compliant lever mechanism yields a model of very small size exhibiting high accuracy.

Keywords: compliant mechanisms, flexure hinges, geometrical nonlinearity, dynamic modeling, model order reduction.

1 Introduction
A new approach to develop a feed unit of ultra-precise machine tools for small workpieces is based on the application of compliant mechanisms (CM) [20]. CMs are monolithic structures incorporating elastically deformable flexure hinges (FH) which are short members of low-stiffness providing relative motion between two adjacent structures of high stiffness through their elastic deformation. Figure 1 illustrates the prototype of a feed unit based on a CM for ultra-precise positioning which is actuated by piezo-electric actuators providing a planar motion of the end effector (EE). The elastic deformation is enabled by the local decrease of the bending stiffness via diminishment of the cross-section [10]. Because FHs are more accurate, simply scalable, cleaner, less noisy and cheaper in manufacturing and maintenance compared to classical pin joints [10], they are predestined especially for high precision, small scale applications like medical devices [13], micro grippers [22] and positioning stages [12]. However, certain disadvantages, namely limited rotation capability due to stress concentration, topology generation of CMs and limitation due to fatigue are identified. Currently, non-intuitive design and optimization techniques are in the main focus of research as well as modeling and controlling strategies [21].

A brief summary of advantages and disadvantages of common modeling approaches is given in Table 1 to motivate the novel approach for dynamic modeling of CM stated in this work. The abbreviation TBT denotes the Timoshenko beam theory which is explained in [18] for beams with functionally variable cross sections and PRBM relates to the pseudo-rigid-body model, see [5, 6, 11] for details. Full 3D finite element methodology (FEM) models of complex CM may easily reach more than 10^6 DOF for mesh independent solutions and therewith are far too large whereas the approximation is highly accurate and nonlinearity may be included. Both finite beam...
and plane elements, as applied in [10], exhibit notable deviations compared to full 3D models as these do not reproduce the accurate deformation state of flexure hinges. The common PRBM approach does not have the ability to fulfill the requirements of ultra precise applications and is not suitable for CMs of complex geometry. The same holds for TBT. To the authors knowledge, accurate and small-scale modeling techniques representing both the static and dynamic response of CMs incorporating nonlinear effects and spatial loading cases are nonexistent, especially for CMs of high complexity.

To embed the CM in a machine tool and drive this mechatronic device by piezo-electric actuators, a controller supplying appropriate input signals is necessary. A model based on FEM of the CM in the form of a linear time invariant (LTI) system constitutes the basis for the controller requiring a low number (< 100) of degrees of freedom (DOF) for real-time operations being as accurate as possible. The approximation of the CM as an elastic multibody system (EMBS) by substructuring in combination with model order reduction (MOR) by component mode synthesis (CMS) and Krylov subspaces fulfills these requirements and offers a scheme of general applicability. This approach will bei described in the forthcoming sections.
2 Significantly deformed domain: a novel modeling approach

In this chapter, a new approach relying on FEM is presented, called significant deformed domain model (SDD), addressing the drawbacks of common modeling procedures discussed in the previous chapter.

First, the method is presented by means of a parabolic FH and is based on several assumptions which are listed below and illustrated in Figure 2. However, the proposed approach is valid for all kinds of FH geometries with concentrated compliance and these assumptions do not limit the range of applicability.

![Figure 2. Geometric parameters defining a FH.](image)

- Two symmetric cutouts of given geometry form the FH.
- The minimal hinge height \( h_0 \), hinge length \( l \), polynomial hinge height \( h(x) = h_0 + \left( \frac{H}{2} - h_0 \right) \left( \frac{x}{L} \right)^p \) of order \( p \), height \( H \), depth \( B \) and length \( L \) define the geometric properties of the system regarded.
- The FH is located at the center of a rectangular block.
- The beam is fixed at one end and free at the other.
- A shear force \( F_z \) is applied at the free end.
- Linear elastic deformations, isotropic material and small displacements are assumed.

The assumption of linear elastic deformations and small strains is applicable investigating single FHs as long as they do not undergo large rotations, as stated in [3]. The assumption of linear deformations and small rotations does not hold for CMs when geometric nonlinearities due to large rotations have to be considered, as explained in Chapter 3.

2.1 Identification of the significantly deformed domain

In order to identify the SDD a linear FEM analysis is performed with ANSYS and the von Mises equivalent stress \( \sigma_{eq} \) is considered. In Figure 3(a) where \( \sigma_{eq} \) is depicted for the parabolic FH described above, it is clearly visible that only a small zone contributes to the highest stress levels while most parts of the structure remain almost stress-free. This is characteristic for FHs with concentrated compliance and motivates the development of a modeling approach dividing the structure into a significantly deformed and a stiff domain via a predefined tolerance value \( t_{SDD} \).

For an arbitrary load \( F_z \) all elements and adjacent nodes in the range of

\[
t_{SDD} \leq \frac{\sigma_{eq, node}}{\sigma_{eq, max}} \leq 1
\]

are selected. In Figure 3, the von Mises stress \( \sigma_{eq} \) is depicted for \( t_{SDD} = 0.00, t_{SDD} = 0.01 \) and \( t_{SDD} = 0.10 \), respectively, showing that the significant region decreases for increasing values of \( t_{SDD} \). A measure for the size of the significant region is defined by \( l_{SDD} \) which is the maximal

255
arising location in the x-direction from the subset of the selected nodes in adjacency to the FH. It should be noted that stresses \( \sigma_{eq} \) occurring on the top and bottom of the adjacent rectangular blocks are not taken into account.

\[
\varepsilon_{eq} = \frac{1}{\sqrt{2(1+\nu)}} \sqrt{(\varepsilon_t - \varepsilon_r)^2 + (\varepsilon_t - \varepsilon_z)^2 + (\varepsilon_z - \varepsilon_y)^2 + \frac{3}{2} (\gamma_{xy}^2 + \gamma_{xz}^2 + \gamma_{yz}^2)},
\]

Figure 3. Distribution of the von Mises stress \( \sigma_{eq} \) for different tolerance values \( t_{SDD} \): highest stresses relate to red color while lowest stresses are indicated by blue color, grey is out of range

The SDD features dependency on the geometric attributes but independency of the absolute load value and linear elastic, isotropic material properties in particular. This results from the one-dimensional Hookean law \( \sigma_{eq} = E\varepsilon_{eq} \) and von Mises strain

Furthermore, the significance of this approach is given by the possibility to determine a substructure representing the essential deformations. FEM calculations show that \( l_{SDD} \) increases as the tolerance value \( t_{SDD} \) decreases and the minimal hinge height \( h_0 \) rises, as depicted in Figure 4. In addition, the length \( l_{SDD} \) may exceed the FH length \( l \) if the relative length of the SDD \( \frac{l_{SDD}}{l} > 1 \) depending on \( t_{SDD} \) and the relative minimal hinge height \( \frac{2h_0}{H} \).

Furthermore, taking \( t_{SDD} = 0.01 \) as a basis, it is found that the geometric parameters \( \frac{l}{H} \) and \( \frac{2h_0}{H} \) have strong impact on \( l_{SDD} \), as depicted in Figure 5(a) where it is clearly visible that \( \frac{l_{SDD}}{l} \) increases for decreasing values of the relative hinge length \( \frac{l}{H} \) and raising values of the relative minimal hinge height \( \frac{2h_0}{H} \), respectively. Again, if \( \frac{l_{SDD}}{l} > 1 \), adjacent structures need to be considered due to accurate modeling.

In order to determine \( l_{SDD} \) independently from FEM calculations, a curve fitting based on the FEM results presented in Figure 5(a) is performed by the MATLAB Curve Fitting Toolbox™ yielding the polynomial function

\[
\frac{l_{SDD}}{l} = 9.34 \left( \frac{2h_0}{H} \right)^8 + 12.16 \frac{2h_0}{H} + 0.27
\]

valid in the interval \( \{ \frac{l}{H} \in \mathbb{R} | 0.1 \leq \frac{l}{H} \leq 1.0 \} \) and \( \{ \frac{2h_0}{H} \in \mathbb{R} | 0.01 \leq \frac{2h_0}{H} \leq 0.1 \} \) which is plotted in Figure 5(b). This procedure exemplary applied to parabolic FHs may be transferred to other types of FH defined by e.g. rectangular, circular or quartic cutouts.
2.2 Modeling of the SDD as superelement

The determination of the SDD described in the previous section enables a novel finite element modeling approach of FHs which overcomes both imprecise results of the widely used PRBM and the numerous of DOF resulting from full 3D FEM approximation. Henceforth, the measure \( l_{SDD} \) is used as radius around the center of the FH to divide the structure with the FH into three subdomains, namely the left and right parts of high stiffness referred to as linkages and the SDD representing the FH. In the following two different treatments of these subdomains are proposed.

In case of a planar CM acting in the \( x-y \)-plane and incorporating thin FHs the linkages are assumed to behave like rigid bodies and therewith may be represented by point masses with specific mass \( (m_i) \), spatial mass centroid location \( (x_{c,i},y_{c,i} \text{ and } z_{c,i}) \), principal moments of inertia about its centroid \( (I_{x,i}, I_{y,i} \text{ and } I_{z,i}) \) and corresponding angles of the principle axes \( (\theta_{x,i}, \theta_{y,i} \text{ and } \theta_{z,i}) \). According to this data which may be retrieved from the FEM or CAD software point masses \( n_{m,i} \) are implemented. The SDD is discretized by 3D structural solid elements. To be able to connect the SDD with the point masses, two pilot nodes \( n_{p,i} \) are inserted at \( x = \pm \frac{l_{SDD}}{2}, y = z = 0 \) being rigidly coupled with all cross-sectional nodes in the intersecting plane of the SDD. The pilot nodes and point masses possess three translational and three rotational DOF, whereas the nodes of the structural solid elements exhibit only three translational DOF. Depending on their nodal location relative to the pilot node, its rotations are internally converted into translations by multipoint constraint (MPC) equations assuring the kinematic coupling. Now, the number of DOF of the SDD

\[
\frac{l_{SDD}}{l_{SDD}} = \frac{2h_0}{l_{SDD}}\]

Figure 5. \( \frac{l_{SDD}}{l_{SDD}} = \frac{2h_0}{l_{SDD}} \) of a parabolic hinge for \( l_{SDD} = 0.01 \)
is reduced using the component mode synthesis (CMS), which is implemented in most FEM software packages and described in [19], yielding a reduced stiffness $K_{SDD}$ and mass matrix $M_{SDD}$ of the SDD. These matrices specify an arbitrary element with undefined geometry relating two nodes $n_{p,1}$ and $n_{p,2}$ with six DOF each, allowing easy connectivity with the point masses. It must be kept in mind that the connection of the pilot nodes with the point masses needs to be capable of geometrically nonlinear effects. The resulting elastic multibody system (EMBS) with rigid linkages named $SDD^{rigid}$ model of a single FH is displayed in Figure 6(a) where additional, massless nodes $n_a$ at the beginning and end of the beam are inserted to apply boundary conditions.

To improve the proposed methodology, the deformation of the linkages are now taken into account yielding general applicability and highly accurate models. The approach is similar to the one previously described but instead of inserting point masses to represent the linkages these are treated in the same way as the SDD. Accordingly, the linkages are also discretized by 3D structural solid elements and pilot nodes $n_{p,i}$ are added and connected as explained before. Applying CMS reduction gives rise to the mass matrices $M_\ell$ and stiffness matrices $K_\ell$ of the flexible linkages. Via the pilot nodes $n_{p,i}$, linkages and SDD may simply be coupled. Figure 6(b) illustrates the $SDD^{elastic}$ approximation by a reduced elastic multibody system (REMSBS).

![Diagram](image)

**Figure 6.** Significant deformed domain model (SDD)

The influence of $t_{SDD}$ on the resulting tip deflection $u_z(x = L)$ is shown in Figure 7 by means of the error

$$
\varepsilon^{u_z} = \frac{u_z(t_{SDD}, \frac{2h_0}{T}) - u_{z, full}(\frac{2h_0}{T})}{u_z(t_{SDD}, \frac{2h_0}{T})} \cdot 100\%.
$$

Obviously, the error $\varepsilon^{u_z}$ rises from a minimal value of -0.2% with both increasing values of $t_{SDD}$ and $\frac{2h_0}{T}$ and reaches up to -15.8%. Furthermore, Figure 7 offers valuable clues to the applicability of the proposed procedure. Regarding $t_{SDD} = 0.01$, errors less than 5% occur up to $\frac{2h_0}{T} = 0.07$ and a maximal value of -9.7% is reached in case of the $SDD^{rigid}$ model. The errors of the $SDD^{elastic}$ model lie in the range of -2.0% and 1.4% and therewith are clearly smaller.

A comparison of the first eigenfrequency $\omega_1$ for full FEM, $SDD^{rigid}$ and $SDD^{elastic}$ depending on the relative minimal hinge heights $\frac{2h_0}{T}$ is listed in Table 2 with corresponding DOF. While the DOF may be importantly reduced by approximately 99.96%, an error of maximal 6% occurs for $SDD^{rigid}$. Because the rigidly assumed linkages do not contribute to the overall deflection this effects higher errors for thickening FHs for which the deformation of the linkages should not be neglected. In addition, it is evident that eigenmodes as well as loadings out of the $x-y$-plane are not accurately approximated by the $SDD^{rigid}$. Regarding the $SDD^{elastic}$ it is clearly visible that errors of less than 1.16% are achieved requiring only 54 DOF.

The advantages of the $SDD^{elastic}$ approach are revealed in the next section where an exemplary compliant lever mechanism is validated.
Figure 7. Resulting error $\varepsilon_{\text{err}}$ depending on the relative minimal hinge height $2h_{\text{f}}/H$ and tolerance value $t_{\text{SDD}}$ for $SDD_{\text{rigid}}$ and $SDD_{\text{elastic}}$, respectively

3 Geometrically nonlinear model of an exemplary compliant mechanism

The proposed modeling method enables easy connection of the SDD models with linkages to form a CM which is exemplarily shown in Figure 8. In a first step, for every FH of different geometric shape appearing in the CM the corresponding SDD is determined as previously explained and the MOR procedure is applied to the SDD. Then the reduced model is inserted at appropriate positions. Secondly, the approximation of the flexible linkages with according pilot nodes are generated and connected to the FHs. Finally, boundary conditions and loads are applied.

To validate the $SDD_{\text{elastic}}$ approach, analysis of an exemplary CM is performed. All calculations of full FEM models are carried out in ANSYS Workbench with automatic, adaptive meshing whereas ANSYS classic via ANSYS Parametric Design Language (APDL) scripts are used to generate and solve the $SDD_{\text{elastic}}$ model. In order to characterize the CM and to obtain deviations of the $SDD_{\text{elastic}}$, a full model meshed with 3D, eight node structural solids (SOLID186) is set up as reference.

Figure 8. Exemplary 3D compliant lever mechanism: Full FEM model with eight quartic flexure hinges meshed by 3D structural solids, fixation at the left, loading $F_x$ and position of the end effector (EE)

As an example, the compliant lever mechanism in Figure 8 is set up to investigate the proposed modeling technique in a technically relevant case and clearly testify nonlinear effects. It is made of the aluminum alloy HOKOTOL ($E = 71$GPa, $v = 0.35$, $\rho = 2830 \text{kg/m}^3$ and $R_{\text{p0.2}} = 535$MPa) and consists of eight thin FHs with fourth order polynomial shape ($p = 4$, $g = 2$mm, $h_0 = 0.1$mm, $h_s = 5$mm $b = 10$mm, $t_{\text{SDD}} = 0.01$ and $x_{\text{SDD}} = 1.60$mm).

The displacements of the end effector EE due to a loading $F_x = 8$N are calculated in a linear as
Table 2. First eigenfrequency $\omega_1$, number of nodes and deviation for parabolic shaped FHs of the full model and corresponding SDD with $t_{SR} = 0.01$

<table>
<thead>
<tr>
<th>$\frac{2h_e}{T_f}$ $[\cdot 10^{-2}]$</th>
<th>Full FEM</th>
<th>$SDD^{rigid}$</th>
<th>$SDD^{elastic}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>DOF</td>
<td>$\omega_1$ (Hz)</td>
<td>DOF</td>
<td>$\omega_1$ (Hz)</td>
</tr>
<tr>
<td>1</td>
<td>$\sim 50,000$</td>
<td>18</td>
<td>20.12</td>
</tr>
<tr>
<td>2</td>
<td>$\sim 50,000$</td>
<td>18</td>
<td>46.71</td>
</tr>
<tr>
<td>3</td>
<td>$\sim 50,000$</td>
<td>18</td>
<td>75.90</td>
</tr>
<tr>
<td>4</td>
<td>$\sim 50,000$</td>
<td>18</td>
<td>106.83</td>
</tr>
<tr>
<td>5</td>
<td>$\sim 50,000$</td>
<td>18</td>
<td>138.33</td>
</tr>
<tr>
<td>6</td>
<td>$\sim 50,000$</td>
<td>18</td>
<td>171.08</td>
</tr>
<tr>
<td>7</td>
<td>$\sim 50,000$</td>
<td>18</td>
<td>205.09</td>
</tr>
<tr>
<td>8</td>
<td>$\sim 50,000$</td>
<td>18</td>
<td>239.16</td>
</tr>
<tr>
<td>9</td>
<td>$\sim 50,000$</td>
<td>18</td>
<td>273.32</td>
</tr>
<tr>
<td>10</td>
<td>$\sim 50,000$</td>
<td>18</td>
<td>308.06</td>
</tr>
</tbody>
</table>

Table 3. Deflection of the EE $u_x$ and $u_z$ for both linear and nonlinear calculation, first four eigenfrequencies and number of nodes for the $SDD^{elastic}$ and the full FEM model of the exemplary CM

<table>
<thead>
<tr>
<th>DOF</th>
<th>$u_x^{lin}$ [mm]</th>
<th>$u_x^{alim}$ [mm]</th>
<th>$u_y^{lin}$ [mm]</th>
<th>$u_y^{alim}$ [mm]</th>
<th>$\omega_1$ [Hz]</th>
<th>$\omega_2$ [Hz]</th>
<th>$\omega_3$ [Hz]</th>
<th>$\omega_4$ [Hz]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full FEM</td>
<td>412 038</td>
<td>-0.96</td>
<td>-1.15</td>
<td>1.67</td>
<td>1.83</td>
<td>38.15</td>
<td>85.57</td>
<td>299.24</td>
</tr>
<tr>
<td>$SDD^{rigid}$</td>
<td>132</td>
<td>-0.94</td>
<td>-1.11</td>
<td>1.62</td>
<td>1.77</td>
<td>38.68</td>
<td>86.78</td>
<td>359.40</td>
</tr>
<tr>
<td>$SDD^{elastic}$</td>
<td>282</td>
<td>-0.95</td>
<td>-1.12</td>
<td>1.64</td>
<td>1.79</td>
<td>38.47</td>
<td>86.29</td>
<td>306.93</td>
</tr>
</tbody>
</table>

well as a nonlinear static analysis for the full FEM model, $SDD^{rigid}$ and the $SDD^{elastic}$. The results are quoted in Table 3. Performing a modal analysis, the first four eigenfrequencies are listed in addition. Besides the remarkable decreased number of DOF by 99.93%, an approximation with inconsiderable deviations of less than 3.23% is reached for the complex shaped CM covering the motion up to the fourth eigenfrequency $f_4 = 1122.00$ Hz for the $SDD^{elastic}$. Moreover, the geometrically nonlinear effects are exactly captured by the $SDD^{elastic}$ model by deviations of less than 2.65%. The non-negligible impact on the results incorporating large rotations yields errors up to 18.80% between linear and nonlinear calculations for the full FEM model regarding $u_x$.

The benefits of the $SDD^{elastic}$ modeling approach over common modeling procedures are listed below.

- Precise modeling of FHs and CMs is allowed with significantly less DOF compared to the FEM model.
- The proposed modeling approach features general applicability to all FHs and CMs.
• Geometrically nonlinear influences are accurately captured.

• The spatial behavior is reproduced with marginal errors considering the elastic deformation of the linkages.

• Once generated, the SDD model of a FH is able to serve as a master model which may be stored in a data base and can be easily implemented to assemble CMs.

Considering the CM of the feed unit in Figure 1, it is obvious that mechanisms of such a high complexity require a considerably increased number of DOF resulting in a model unfeasible for realtime control purposes. This motivates the application of an adapted model order reduction (MOR) technique to provide a linear model of small scale to be implemented in a linear controller.

4 Projective model order reduction by block Krylov subspaces of proportionally damped second order MIMO systems

Though the application of the $SDD^{\text{elastic}}$ reduces the total number of unknowns in the resultant system of equations significantly, a further reduction of the number of unknowns is necessary in order to meet the requirements of a linear control system. Linear model order reduction (MOR) aims at the approximation of linear dynamic systems on the basis of well-established mathematical theories and robust numerical algorithms to provide reduced order models (ROMs) with considerably less DOF and preferably minor errors. On this basis, the computing time of simulation, optimization and control may effectively be shortened.

Various methods for linear and nonlinear dynamical systems are available, for instance modal condensation, balanced truncation approximation and proper orthogonal decomposition. An overview can be found in [1], [9] and [15]. Basically, the following demands are addressed in this work to assure applicability of the model for linear control purposes:

• Create a model as small as possible.

• Assure a small approximation error.

• Preserve the input-output behavior.

• Keep the structure of a second order system.

• Apply efficient and numerically stable algorithms.

In this work, an implicit moment matching procedure for second order multiple input multiple output (MIMO) systems based on Krylov subspaces is utilized which preserves the second order structure for further utilization in elastic multibody dynamics, see e.g. [9]. Details may be found in [2], [8], [14], [16] and [17] where also extensions to adaptive, interpolative and iterative methods are presented. In this work, MOR by Krylov subspaces is solely used as a tool to gain system matrices of small size for CMs. The choice for Krylov subspace reduction is based on the fact that methods based on singular value decomposition like balanced truncation and Hankel norm approximation are restricted to small scale problems since the Lyapunov equations need to be solved and therewith are computationally expensive as summarized in [15]. Even though proper orthogonal decomposition has a wide range of linear and nonlinear applications the adequate choice of snapshots, which needs to be provided, has adverse effects. A more detailed summary of advantages and disadvantages of different MOR approaches may be found in [15].

It is assumed that the dynamic behavior of a CM arising from spatial discretization may be represented by a second order linear time invariant system of order $n$

$$\mathcal{H}_n = \begin{cases} \mathbf{M} \ddot{\mathbf{q}}(t) + \mathbf{D} \dot{\mathbf{q}}(t) + \mathbf{K} \mathbf{q}(t) = \mathbf{B} \mathbf{u}(t) \\ \mathbf{y}(t) = \mathbf{C} \mathbf{q}(t) \end{cases}$$ (5)
with the mass matrix $\mathbf{M} \in \mathbb{K}^{n \times n}$, damping matrix $\mathbf{D} \in \mathbb{K}^{n \times n}$, stiffness matrix $\mathbf{K} \in \mathbb{K}^{n \times n}$, input matrix $\mathbf{B} \in \mathbb{K}^{n \times m}$, output matrix $\mathbf{C} \in \mathbb{K}^{q \times n}$, the internal state vector $\mathbf{q}(t) \in \mathbb{K}^n$, input vector $\mathbf{u}(t) \in \mathbb{K}^n$ and output vector $\mathbf{y}(t) \in \mathbb{K}^q$ having multiple inputs and multiple outputs (MIMO), as described in [14].

The aim is to provide a reduced second order system of order $r \ll n$

$$
\mathcal{J}_r = \left\{ \mathbf{M}_r \dot{\mathbf{q}}_r(t) + \mathbf{D}_r \mathbf{q}_r(t) + \mathbf{K}_r \mathbf{q}_r(t) = \mathbf{B}_r \mathbf{u}(t) \right\}
$$

by applying the projection matrices $\mathbf{V} \in \mathbb{K}^{n \times r}$ and $\mathbf{W} \in \mathbb{K}^{n \times r}$ to Equation 5 yielding the reduced matrices $\mathbf{M}_r = \mathbf{W}^T \mathbf{M} \mathbf{V}$, $\mathbf{D}_r = \mathbf{W}^T \mathbf{D} \mathbf{V}$, $\mathbf{K}_r = \mathbf{W}^T \mathbf{K} \mathbf{V}$ and $\mathbf{B}_r = \mathbf{W}^T \mathbf{B} \in \mathbb{K}^{r \times m}$, $\mathbf{C}_r = \mathbf{C} \mathbf{V} \in \mathbb{K}^{q \times r}$. The task of MOR is to determine the projection matrices $\mathbf{W}$ and $\mathbf{V}$.

The numerical implementation to compute the projection matrices is carried out by the two-sided rational Arnoldi interpolation algorithm for proportionally damped second-order MIMO systems referring to [9, 16]. Note that the size of the ROM also depends on the number of inputs and outputs, respectively.

5 Reduced order, linear model of a compliant mechanism

The above described MOR procedure is applied to the $SDD^{\text{elastic}}$ of the compliant lever mechanism described before. Reduced models of different sizes are generated and the errors of the frequency responses are compared. For simplicity reason, damping is not taken into account. Besides the system matrices $\mathbf{M}$ and $\mathbf{K}$, the inputs and outputs forming the matrices $\mathbf{B}$ and $\mathbf{C}$ have strong impact on the size of the ROM.

The process of reducing the system size incorporates the following steps and software:

1. Generate $SDD^{\text{elastic}}$ using ANSYS APDL $\rightarrow \mathbf{M}$ and $\mathbf{K}$.
2. Specify input and output at corresponding nodes $\rightarrow \mathbf{B}$ and $\mathbf{C}$.
3. Import FE data into MatMorembas [4].
4. Perform MOR with MatMorembas $\rightarrow \mathbf{M}_r, \mathbf{K}_r, \mathbf{B}_r$ and $\mathbf{C}_r$.
5. Calculate and plot error regarding frequency responses.
6. Write and export reduced system matrices.
7. Perform simulations, control and optimizations.

The results of the reduced model $SDD^{\text{elastic}}_{\text{red}}$ are evaluated by the relative approximation error of the frequency response using the Frobenius-norm

$$
\varepsilon_{\text{MOR}}(f) = \frac{\| \mathbf{H}(f) - \mathbf{H}_r(f) \|_F}{\| \mathbf{H}(f) \|_F},
$$

in the range of $0 \leq f \leq 1200$Hz being depicted in Figure 9 where also the first four eigenfrequencies at $\omega_1 = 38.21$Hz, $\omega_2 = 85.72$Hz, $\omega_3 = 356.63$Hz and $\omega_4 = 1158.20$Hz are marked. For comparison, the result for modal condensation of size $r = 6$ is given. Thus it appears that a Krylov ROM of the CM with significantly lower dimension may be specified. An error of $\varepsilon_{\text{MOR}} < 10^{-8}$ is achieved for $r = 12$ over the examined frequency range indicating the preservation of the linear input-output behavior and for this reason an efficient approximation of the $SDD^{\text{elastic}}$. Therewith, small-scale but highly accurate linear models are made available for the simulation and control of CMs. In accordance with the fact that a linear model is desired to be implemented in an efficient, real-time linear controller linear MOR approaches are applied.
6 Conclusions

This paper presents a novel methodology to generate small, but highly accurate models of compliant mechanisms. In a first step, the significantly deformed region of a CM is taken into account to generate a compact model, called $SDD_{\text{elastic}}$ model which incorporates superelements to keep the important system characteristics and be capable of large, nonlinear deformations without significant loss of accuracy. In a second step, MOR using Krylov subspace reduction for proportionally damped second order MIMO systems is used for further decrease of the system matrices defined by the $SDD_{\text{elastic}}$.

Highly accurate, linear models of very small size with $r = 12$ DOF for control purposes are gained exhibiting a marginal error. This approach of first generating the $SDD_{\text{elastic}}$ of a CM and then reducing it by means of Krylov subspace reduction avoids the intense computation and reduction of full CM models and is generally practicable for efficient modeling of CMs. Depending on the intended use, both nonlinear and linear models are provided.

Acknowledgement

This research was financially supported within the priority program SPP1476 "Small machine tools for small workpieces" from the Deutsche Forschungsgemeinschaft (DFG). The authors thank their colleagues at the Institute of Technical and Numerical Mechanics at the University of Stuttgart for providing software in the field of model order reduction.

REFERENCES


Compatibility Equation and Revision on the Expression of Chord-line Component of Dynamical Cable Tension

Guo-Kang Er, Kun Wang, Vai Pan Iu

Department of Civil and Environmental Engineering
University of Macau
Macau SAR, China
yb37402@umac.mo

ABSTRACT
The problem in deriving the relation of the chord-line component of the dynamical cable tension and the deflection in the conventional cable theory is investigated. The expression for the relation of the chord-line component of dynamical cable tension and the deflection is reasonably derived and improved by introducing a compatibility condition for vibrations of cable. The difference between the chord-line component of dynamical cable tension based on the conventional cable theory and the proposed compatibility equation are investigated for different sag-to-span ratios and inclined angles. It is found that the error caused by the conventional cable theory can be too large to be acceptable when the sag-span ratio or the inclined angle is large to some extent.

Keywords: Cable vibration, Compatibility equation, Chord-line tension.

1 Introduction
Cables are nowadays widely used in many long-span civil engineering structures and towering mechanical systems, such as cable-stayed bridges, suspension bridges, and overhead power transmission line, and others. The research about cables has developed for several decades. In analyzing cable vibrations, the influence of geometrical nonlinearity can not be ignored. Otherwise, the analytical results can not reflect the real behavior of the cable. In 1970s, Irvine and Caughey developed the nonlinear theory for the static analysis of suspended extensible cables and the linear theory for the free vibrations of suspended extensible horizontal cable in which the ratio of sag to span is about 1/8 or less and the vibrations of an inclined cable by neglecting the weight component parallel to the cable chord [1, 2]. Hagedorn and Schafer extended the linear theory of cable vibration by considering the effect of quadratic and cubic nonlinear terms in 1980 [3]. They analyzed the nonlinear vibration of the cable numerically using the first normal mode of the linear elastic flexible cable as shape function. Luongo et al developed the nonlinear equations of motion of elastic cable about deformed static equilibrium configuration by means of Hamilton’s principle in 1984 [4]. The planar motion is described by the differential equation of transverse displacement through neglecting the longitudinal inertia. Using a single-degree-of-freedom model, Benedettini and Rega analyzed the in-plane response caused by harmonic load uniformly distributed on cable in 1987 [5]. Perkins investigated the nonlinear response of a suspended elastic cable due to planar excitation by first-order perturbation analysis in 1992 [6]. The existence of nonlinear terms makes the in-plane cable motion coupled with the out-of-plane cable motion and induces modal interactions. Benedettini et al investigated the coupled vibrations of the elastic suspended cable subjected to both external forces and support motions using a four-degree-of-freedom model with perturbation method in 1995 [7]. Rega reviewed the mechanical and mathematical theories of the elastic suspended cables and the methods used for analyzing cable vibration problems in 2004 [8, 9]. Wu et al studied the in-plane motion of an inclined cable in the local coordinate system and the modification of the expressions for the in-plane natural frequencies of an inclined cable was presented in 2005 [10]. Georgakis and Taylor presented an extensive study on the vibrational behavior of the cable with small sag-to-span ratio under the action of sinusoidal and stochastic loads in cable plane in 2005 [11, 12]. The cables with both in-plane and out-of-plane displacements were analyzed.
All the above analysis are based on the conventional theory about cable analysis. It is found that there is a problem in deriving the relation of the chord-line component of the dynamical cable tension and the deflection in the conventional cable theory [13, 14]. In some publications the chord-line component of the dynamical cable tension is expressed as a function of both space and time, but not a function of time alone when the load is perpendicular to the chord line [1, 2]. In some other publications the chord-line component of dynamical cable tension is given from the average strain in the whole length of the cable [4, 6, 7, 8]. Both of these two expressions are not reasonably derived. Hence large error can be produced in some cases. The objectives of this paper are to present the revised relation of the chord-line component of the dynamical cable tension and the deflection obtained by the way of compatibility equation for in-plane vibrations of the cable. The difference between the chord-line components of the dynamical cable tension calculated based on the conventional theory and the presented theory are investigated in detail with numerical results for different sag-to-span ratios and inclined angles.

2 Equations of Motion of an Inclined Supported Suspended Cable

The analysis of cable is based on the following two assumptions: (1) The flexural rigidity, torsional rigidity, and shear stiffness of the cable are neglected. (2) The deformation constitutive relation of the cable satisfies the Hooke’s law and the stress on the cross section of cable is uniform.

2.1 Vertical static equilibrium

Consider a uniform inclined suspended cable with initial pretension and inclined angle $\theta$ hanging on two supports in its ends as shown in Figure 1. The transformation between the coordinate systems $x'Oy'$ and $xOy$ is $x = x'\cos \theta + y'\sin \theta$ and $y = -x'\sin \theta + y'\cos \theta$. The equations for static equilibrium of the cable under the coordinate system $x'Oy'$ can be formulated as follows based on Figure 2.

\[
\sum F_x = 0 : \frac{\partial}{\partial s} \left( T \frac{\partial y'}{\partial s} \right) = 0
\]

\[
\sum F_y = 0 : \frac{\partial}{\partial s} \left( T \frac{\partial y'}{\partial s} \right) = -\rho g
\]
where \( y' \) is the static displacement in \( y' \) direction due to both the self-weight of the cable and the pretension in the cable, \( T \) is the tension in the cable in static state, \( s \) is the coordinate along the cable length in static state, \( \rho \) is the mass per unit length of the cable. Denote the horizontal component of the cable tension at static state as \( H' \) and the inclined length between two supports as \( l \).

Assume that when \( x' = a \), the slope of the profile equals to zero, that is \( \frac{dy'}{dx'} \big|_{x'=a} = 0 \). The solution of Equation (2) can be obtained with the boundary condition \( y' = 0 \) at \( x' = 0 \) and \( y' = l \sin \theta \) at \( x' = l \cos \theta \) as follows.

\[
y' = -\frac{H'}{\rho g} \cosh \left[ -\frac{\rho g}{H'} (x' - a) \right] + \frac{H'}{\rho g} \cosh \left( \frac{\rho g a}{H'} \right) \tag{3}
\]

with the following equation for determining the value of \( a \).

\[
-\frac{H'}{\rho g} \cosh \left[ -\frac{\rho g}{H'} (l \cos \theta - a) \right] + \frac{H'}{\rho g} \cosh \left( \frac{\rho g a}{H'} \right) = l \sin \theta \tag{4}
\]

As shown in Figure 1, the profile of the line between two supports is \( y'_1 = x' \tan \theta \), so

\[
y'_2 = y' - y'_1 = -\frac{H'}{\rho g} \cosh \left[ -\frac{\rho g}{H'} (x' - a) \right] + \frac{H'}{\rho g} \cosh \left( \frac{\rho g a}{H'} \right) - x' \tan \theta
\]

Letting \( \frac{dy'_2}{dx} = 0 \), it gives that when \( x' = a - \frac{H'}{\rho g} \text{asinh}(\tan \theta) \), \( y'_2 \) reaches its maximum value given by

\[
y'_{2,max} = -\frac{H'}{\rho g} \cosh[\text{asinh}(\tan \theta)] + \frac{H'}{\rho g} \cosh \left( \frac{\rho g a}{H'} \right) - \tan \theta \left[ a - \frac{H'}{\rho g} \text{asinh}(\tan \theta) \right] \tag{5}
\]

So the maximum sag can be expressed as \( d = y'_{2,max} \cos \theta \). For given allowable value of sag-span ratio \( d/l \), the value of \( y'_{2,max} \) can be determined and then the value of \( H' \) and \( a \) can be obtained with Equations (4) and (5).

The equations for static equilibrium of the cable under the coordinate system \( xOy \) can be formulated as follows based on Figure 3.

\[
\sum F_x = 0 : \frac{\partial}{\partial s} \left( T \frac{\partial x}{\partial s} \right) = -\rho g \sin \theta \tag{6}
\]

\[
\sum F_y = 0 : \frac{\partial}{\partial s} \left( T \frac{\partial v_0}{\partial s} \right) = -\rho g \cos \theta \tag{7}
\]

where \( v_0 = y \) is the static displacement in \( y \) direction due to both the self-weight of the cable and the pretension in the cable.
2.2 One dimensional motion

If the inclined cable is subjected to external dynamical force \( f_y(x,t) \) per unit length in \( y \) direction. The dynamical equilibrium equations are given as follows based on Figure 4.

\[
\sum F_x = 0 : \frac{\partial}{\partial s} \left[ (T + \tau) \frac{\partial (x + u)}{\partial s} \right] = \rho \frac{\partial^2 u}{\partial t^2} + c_1 \frac{\partial u}{\partial t} - \rho g \sin \theta \quad (8)
\]

\[
\sum F_y = 0 : \frac{\partial}{\partial s} \left[ (T + \tau) \frac{\partial (v_0 + v)}{\partial s} \right] = \rho \frac{\partial^2 v}{\partial t^2} + c_2 \frac{\partial v}{\partial t} - \rho g \cos \theta - f_y(x,t) \quad (9)
\]

where \( \tau \) is the additional tension generated by displacements \( u \) and \( v \) which are the longitudinal and transverse dynamical displacements, respectively; \( f_y(x,t) \) is the distributed dynamical force perpendicular to the chord line; \( c_1 \) and \( c_2 \) are damping coefficients.

If there is no dynamical force in \( x \) direction, the longitudinal dynamical displacement \( u \) can be neglected because it is much small compared with \( v \). Then the in-plane or two-dimensional cable vibration is reduced to one dimensional cable vibration. From the geometry of the cable, it is known that

\[
ds = (1 + y'^2) \frac{dx}{ds} \quad (10)
\]

\[
ds' = (1 + y'^2) \frac{dx'}{ds'} \quad (11)
\]

\[
H' = \frac{T}{ds} \quad (12)
\]

\[
h = \tau \frac{dx}{ds} \quad (13)
\]

where \( y \) and \( y' \) denote the derivatives of \( y \) and \( y' \) with respect to \( x \) and \( x' \), respectively; \( h \) is the chord-line component of the additional cable tension induced by dynamical displacement. It is a function of time alone when the cable is subjected to the dynamical force \( f_y(x,t) \).

Introducing Equation (7) into Equation (9) and using Equations (10), (11), (12), (13), the equation of motion of the one dimensional cable vibration is given as follows.

\[
\rho \frac{\partial^2 v}{\partial t^2} + c_2 \frac{\partial v}{\partial t} - \frac{\partial}{\partial s} \left[ H' \frac{dx}{dx'} \frac{\partial v}{\partial x} \right] - (1 + y'^2)^{-\frac{1}{2}} \frac{\partial}{\partial x} \left[ H \left( \frac{\partial v}{\partial x} + \frac{dv}{dx} \right) \right] = f_y(x,t) \quad (14)
\]

where

\[
\frac{\partial}{\partial s} \left( H' \frac{dx}{dx'} \frac{\partial v}{\partial x} \right) = H' \frac{\partial}{\partial s} \left( \frac{dx}{dx'} \right) \frac{\partial v}{\partial x} + H' \frac{dx}{dx'} \frac{\partial}{\partial s} \left( \frac{\partial v}{\partial x} \right)
\]

\[
= H'(1 + y'^2)^{-\frac{1}{2}} \frac{\partial}{\partial x} \left( \frac{dx}{dx'} \right) \frac{\partial v}{\partial x} + H' \frac{dx}{dx'} \frac{\partial}{\partial x} \left( \frac{\partial v}{\partial x} \right)
\]

and

\[
\frac{dx}{dx'} = \frac{\partial x}{\partial x'} + \frac{\partial x}{\partial y'} \frac{dy'}{dx'} = \cos \theta + \sin \theta \frac{dy'}{dx'}
\]
3 Problems about Chord-line Component of Dynamical Cable Tension in Conventional Cable Theory

In Equation (14), the expression of $h$ which is the chord-line component of the dynamical cable tension is needed. However, a problem is found in the conventional cable theory in deriving the expression of $h$.

3.1 Conventional expression 1 for the $h$ in 1D cable vibration

One conventional expression of $h$ for the 1D vibration of suspended cables without considering excitation along chord-line can be found in [1]. Let $ds$ denote the length of the differential element at static state and $ds'$ denote the length of the differential element in dynamical state. Denote the strain due to dynamical deformation as $\varepsilon$. Then

$$
(ds)^2 = (dx)^2 + (dy)^2.
$$

$$
(ds')^2 = (dx + du)^2 + (dy + dv)^2.
$$

$$
\varepsilon = \frac{ds'}{ds} - \frac{ds}{ds}.
$$

Substituting Equation (15) into (16) and expressing $\varepsilon$ with Taylor series and remaining the terms up to second order, it gives

$$
\varepsilon(s,t) \approx \frac{dx du}{ds ds} + \frac{dy dv}{ds ds} + \frac{1}{2} \left( \frac{du}{ds} \right)^2 + \frac{1}{2} \left( \frac{dv}{ds} \right)^2
$$

If the term $\frac{1}{2} \left( \frac{du}{ds} \right)^2$ is small compared to other terms, it can be neglected. Then

$$
\varepsilon(s,t) \approx \frac{dx du}{ds ds} + \frac{dy dv}{ds ds} + \frac{1}{2} \left( \frac{dv}{ds} \right)^2
$$

With Hooke’s law,

$$
\tau = EA \cdot \varepsilon,
$$

where $E$ is the Young’s modulus of the cable and $A$ is the area of cable cross section. Substituting Equations (13) and (18) into Equation (19), it gives

$$
\frac{h}{EA} \left( \frac{ds}{dx} \right)^3 = \frac{du}{dx} + \frac{dy dv}{dx dv} + \frac{1}{2} \left( \frac{dv}{dx} \right)^2
$$

From Equation (20) it is seen that $h$ is a function of both $x$ and $t$. It is not reasonable because $h$ equals the chord-line component of the dynamical cable tension at supports and it should be free of $x$. One contradictory case is given in the following. Under the action of the force $f_y$ uniformly distributed on the cable and perpendicular to the chord line, the deformation of the cable gives that $\frac{dy}{dx}$, $\frac{dv}{dx}$, and $\frac{ds}{dx}$ in Equation (20) equal zero at $x = \frac{l}{2}$, which leads to $h = 0$ at $x = \frac{l}{2}$. This does not agree with the fact that $h$ is nonzero at supports if $f_y$ is nonzero. Hence Equation (20) is contradictory to the fact that $h$ is free of $x$.

In order to make $h$ free of $x$, Equation (20) is integrated on both sides from 0 to $l$ in conventional cable theory. With the boundary conditions $u(0) = u(l) = 0$, it gives

$$
h = \frac{EA}{L_e} \int_0^l \left[ \frac{dy}{dx} + \frac{1}{2} \left( \frac{dv}{dx} \right)^2 \right] dx
$$

where

$$
L_e = \int_0^l \left( \frac{ds}{dx} \right)^3 dx = \int_0^l (1 + y^2)^{3/2} dx
$$

It is not a rational step from Eq. (20) to Eq. (21).
3.2 Conventional expression 2 for the in 1D cable vibration

The other conventional expression of \( h \) can be found in [4, 6, 7]. For the in-plane vibrations of suspended cable without excitation along chord line, the cable strain is given as follows.

\[
(ds)^2 = (dx)^2 + (dy)^2
\]

\[
(ds')^2 = (dx + du)^2 + (dy + dv)^2
\]

\[
\varepsilon(s,t) = \frac{ds' - ds}{ds} \approx \frac{dx}{ds} \frac{du}{ds} + \frac{dy}{ds} \frac{dv}{ds} + \frac{1}{2} \left( \frac{dv}{ds} \right)^2. \tag{24}
\]

The equations of motion for the in-plane vibrations of the cable can be formulated with Hamilton’s principle. Using \( ds \approx dx \) and neglecting the gradient of the longitudinal \((u)\) component of the dynamical displacement, the equations of motion are given as follows.

\[
\frac{\partial}{\partial x} \left\{ EA \left[ \frac{\partial u}{\partial x} + dy \frac{\partial v}{\partial x} + \frac{1}{2} \left( \frac{\partial v}{\partial x} \right)^2 \right] \right\} = \rho \frac{\partial^2 u}{\partial t^2} + c_1 \frac{\partial u}{\partial t} \tag{25}
\]

\[
\frac{\partial}{\partial x} \left\{ H \frac{\partial v}{\partial x} + EA \left( \frac{dy}{dx} + \frac{\partial v}{\partial x} \right) \left[ \frac{\partial u}{\partial x} + \frac{dy}{dx} \frac{\partial v}{\partial x} + \frac{1}{2} \left( \frac{\partial v}{\partial x} \right)^2 \right] \right\} - f_y = \rho \frac{\partial^2 v}{\partial t^2} + c_2 \frac{\partial v}{\partial t} \tag{26}
\]

The expression inside \{ \} in Equation (25) is the chord-line component of dynamical cable tension \( h \). Neglecting the longitudinal inertia and viscous forces in Equation (25) gives

\[
\frac{\partial u}{\partial x} + dy \frac{\partial v}{\partial x} + \frac{1}{2} \left( \frac{\partial v}{\partial x} \right)^2 = \varepsilon(t) \tag{27}
\]

The left hand side of Equation (27) is a function of both space and time, but the right-hand side is a function of time only, which is not rational. Integrating both sides of Equation (27) from 0 to \( l \) with the boundary conditions \( u(0) = u(l) = 0 \), it gives

\[
\varepsilon(t) = \frac{1}{7} \int_0^l \left[ \frac{dy}{dx} \frac{dv}{dx} + \frac{1}{2} \left( \frac{dv}{dx} \right)^2 \right] dx \tag{28}
\]

The chord-line component of dynamical cable tension is calculated by \( h = EA \varepsilon(t) \), which is a function of time only. It is also not a rational step from Eq. (27) to Eq. (28).

For the cable with large sag-to-span ratio, \( ds = (1 + y^2)^{\frac{1}{2}} dx \) is used in the derivation and the expression of \( h \) is given as follows.

\[
h = \frac{EA}{L_c} \int_0^l \left[ \frac{dy}{dx} \frac{dv}{dx} + \frac{1}{2} \left( \frac{dv}{dx} \right)^2 \right] dx \tag{29}
\]

where \( L_c \) is the same as that in Equation (22).

4 Compatibility Equation and Revised Expression of \( h \)

In order to derive the expression of \( h \) rationally, a compatibility equation is introduced and formulated as follows. The length change of the cable before and after dynamical deformation satisfies the following compatibility condition.

\[
\Delta L = \int_0^L ds' - \int_0^L ds = \int_0^L \varepsilon ds = \int_0^L \frac{\tau}{EA} ds \tag{30}
\]

Substituting Equations (13) and (23) into Equation (30) gives

\[
\int_0^l \left[ \sqrt{(1 + u^2) + (y + v)^2} - \sqrt{1 + y^2} \right] dx = \frac{1}{EA} \int_0^l \left[ h \cdot \left( \frac{ds}{dx} \right)^2 \right] dx \tag{31}
\]
The left-hand side of Equation (31) is expanded using Taylor series up to second-order terms and neglecting the small term $\frac{1}{2}u_x^2$. It gives the following expression of $h$ with the boundary conditions $u(0) = u(l) = 0$.

$$h = \frac{EA}{L_c} \int_0^l \left( v_xv_x + \frac{1}{2}v_x^2 \right) dx$$  \hspace{1cm} (32)

where

$$L_c = \int_0^l \left( \frac{ds}{dx} \right)^2 dx = \int_0^l (1 + y_x^2) dx$$ \hspace{1cm} (33)

The above derivation leads to the difference between $L_e$ in Equation (22) and $L_c$ in Equation (33). The difference is slight for the cable with small sag-to-span ratio, but it can be obvious for the cable with large sag-to-span ratio. The derivation of Equation (32) is based on the rational compatibility equation (30).

5 Numerical Results and Comparison

Consider the inclined cable with mass per unit length being 4539.6 kg/m, length between two supports being 1000m. In order to calculate the difference between $L_e$ in Equation (22) and $L_c$ in Equation (33), which leads to the difference between the chord-line component of dynamical cable tension calculated based on the conventional cable theory and the proposed compatibility equation, the derivative of $y$ with respect to $x$ is needed and calculated first in the following. The relationship between the coordinate systems $xOy$ and $x'Oy'$, gives

$$x = x' \cos \theta + y' \sin \theta$$

$$y = -x' \sin \theta + y' \cos \theta$$

The derivation of $y$ with respect to $x$ is given as

$$\frac{dy}{dx} = \frac{\partial y}{\partial x'} \cdot \frac{dx'}{dx} + \frac{\partial y}{\partial y'} \cdot \frac{dy'}{dx} = \frac{-\sin \theta + \cos \theta \cdot \sinh \left[ -\frac{\rho g}{H'} (x' - a) \right]}{\cos \theta + \sin \theta \cdot \sinh \left[ -\frac{\rho g}{H'} (x' - a) \right]}$$ \hspace{1cm} (34)

Substituting Equation (34) into Equation (22) and Equation (33), respectively, gives the values of $L_e$ and $L_c$. They are shown and compared for different sag-to-span ratio $d/l$ and inclined angle $\theta$ as shown in the Table 1 to Table 5.

**Table 1.** Difference between $L_e$ and $L_c$ for $\theta = 0^\circ$  **Table 2.** Difference between $L_e$ and $L_c$ for $\theta = 15^\circ$

<table>
<thead>
<tr>
<th>sag-span ratio</th>
<th>$L_e$</th>
<th>$L_c$</th>
<th>$(L_e - L_c)/L_c (%)$</th>
<th>sag-span ratio</th>
<th>$L_e$</th>
<th>$L_c$</th>
<th>$(L_e - L_c)/L_c (%)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/10</td>
<td>1082.34</td>
<td>1053.61</td>
<td>2.73</td>
<td>1/10</td>
<td>1046.6</td>
<td>1018.41</td>
<td>2.77</td>
</tr>
<tr>
<td>1/8</td>
<td>1130.72</td>
<td>1084.01</td>
<td>4.31</td>
<td>1/8</td>
<td>1095.07</td>
<td>1048.76</td>
<td>4.42</td>
</tr>
<tr>
<td>1/6</td>
<td>1240.27</td>
<td>1150.22</td>
<td>7.83</td>
<td>1/6</td>
<td>1208.42</td>
<td>1116.64</td>
<td>8.22</td>
</tr>
<tr>
<td>1/4</td>
<td>1590.94</td>
<td>1343.18</td>
<td>18.45</td>
<td>1/4</td>
<td>1613.65</td>
<td>1330.88</td>
<td>21.25</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>sag-span ratio</th>
<th>$L_e$</th>
<th>$L_c$</th>
<th>$(L_e - L_c)/L_c (%)$</th>
<th>sag-span ratio</th>
<th>$L_e$</th>
<th>$L_c$</th>
<th>$(L_e - L_c)/L_c (%)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/10</td>
<td>777.17</td>
<td>752.22</td>
<td>3.32</td>
<td>1/10</td>
<td>777.17</td>
<td>752.22</td>
<td>3.32</td>
</tr>
<tr>
<td>1/8</td>
<td>832.91</td>
<td>785.71</td>
<td>6.01</td>
<td>1/8</td>
<td>832.91</td>
<td>785.71</td>
<td>6.01</td>
</tr>
<tr>
<td>1/6</td>
<td>1027.74</td>
<td>888.64</td>
<td>15.65</td>
<td>1/6</td>
<td>1027.74</td>
<td>888.64</td>
<td>15.65</td>
</tr>
<tr>
<td>1/4</td>
<td>1819.06</td>
<td>1338.03</td>
<td>35.95</td>
<td>1/4</td>
<td>1819.06</td>
<td>1338.03</td>
<td>35.95</td>
</tr>
</tbody>
</table>
Table 5. Difference between $L_e$ and $L_c$ for $\theta = 60^\circ$

<table>
<thead>
<tr>
<th>sag-span ratio</th>
<th>$L_e$</th>
<th>$L_c$</th>
<th>$(L_e - L_c)/L_c(%)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/10</td>
<td>572.12</td>
<td>545.19</td>
<td>4.94</td>
</tr>
<tr>
<td>1/8</td>
<td>672.36</td>
<td>599.19</td>
<td>12.21</td>
</tr>
<tr>
<td>1/6</td>
<td>2024.96</td>
<td>1017.46</td>
<td>99.02</td>
</tr>
</tbody>
</table>

It can be seen from Table 1 to Table 5 that the difference between $L_e$ and $L_c$ increases as the sag-to-span ratio increases when the inclined angle remains the same. The difference between $L_e$ and $L_c$ increases as the inclined angle increases when the sag-to-span ratio remains the same.

It can be seen from Table 1 that even when the cable is suspended horizontally the difference between $L_e$ and $L_c$ reaches 18.45% if the sag-to-span ratio is 1/4. This difference is too large to be acceptable. From Table 5, it can be seen that when the inclined angle is 60$^\circ$ and the sag-to-span ratio is 1/10, the difference between $L_e$ and $L_c$ could reach about 5%. The value of $L_e$ or $L_c$ can significantly influence the responses of cable vibrations.

6 Conclusions

The problem in the conventional cable theory about the relation between the chord-line component of dynamical cable tension and the deflection is discussed. The compatibility equation is formulated with which the relation between the chord-line component of dynamical cable tension and the deflection is rationally derived for one-dimensional vibration of suspended cable. The expression of the chord-line component of dynamical cable tension is revised. The difference between the parameters $L_e$ calculated based on the conventional cable theory and $L_c$ calculated based on the proposed compatibility equation is discussed for different sag-to-span ratios and inclined angles of cable. The error caused by the conventional cable theory can be too large to be acceptable when the sag-to-span ratio or the inclined angle is large to some extent.

7 Acknowledgement

This research is supported by the Research Committee of University of Macau (Grant No. MYRG138-FST11-EGK, MYRG2014-0084-FST) and the Science and Technology Development Fund of Macau (Grant No. 043/2013/A).

REFERENCES


Deployment dynamics of a large-scale ring truss mesh antennas

Pei Li, Cheng Liu, Qiang Tian, Haiyan Hu

School of Aerospace Engineering
Beijing Institute of Technology
5 South Zhongguancun Street, Haidian District, Beijing, 100081, China
lipei0603@sina.com, liucheng_bit@aliyun.com,
tianqiang_hust@aliyun.com, haiyan_hu@bit.edu.cn

ABSTRACT
The deployment dynamics of a large-scale satellite mesh reflector is studied via Absolute Nodal Coordinate Formulation (ANCF). The large scale multibody system model is firstly decomposed into several independent subsystems by cutting its joints or an ANCF element grid. Then, the static condensation method is used to eliminate each subsystem’s internal generalized coordinates and Lagrange multipliers for joint constraint equations. To reduce the order of these simultaneous linear equations, the subsystems are further divided by using the multilevel decomposition approach. The equations of motion are solved by using an OpenMP based parallel generalized-alpha algorithm. Finally, the dynamics computation for a 500 seconds deployment process of a complex AstroMesh reflector with over 190,000 generalized coordinates is efficiently achieved within 78 hours.

Keywords: mesh reflector, deployment dynamics, Absolute Nodal Coordinate Formulation (ANCF), static condensation, parallel algorithm.

1 INTRODUCTION
Over the past years more and more deployable space antenna reflector with large apertures, high accuracy, light weight and high rigidity has been used to meet the increasing requirements from the telecommunication industry [1]. The deployment of this kind of structures is a complicated transition process from a multibody system with relative motions among its components to a structure, which is one of the most dangerous stages in the spacecraft launching process. In history, an umbrella-type antenna, the Rigid-Rib Antenna (RRA) onboard the Galileo spacecraft failed to unfold as expected in 1991 [2]. One of the reasons for the failure is probably that the dynamics features of the deployment process were not forecasted accurately. Therefore, it is necessary to simulate deployment process during the structure design phase so that a deep insight on the deployment dynamic behaviors can be obtained. Thus, the subject of the deployment dynamics of space structures has attracted many researchers’ attention because of its importance and challenge.

In the framework of the multi-rigid body dynamics, Li [3] studied the deployment dynamics of a rigid ring truss antenna based on the Lagrange Equations of the second kind. However, the simulation based on the multi-rigid body dynamics can not accurately reflect the practical dynamics behaviors of the reflectors with large aperture and light weight because the deformation of their flexible components. Using flexible multibody dynamics theory, Neto and Ambrósio et al. [4] analyzed and optimized the deployment process of a Synthetic Aperture Radar (SAR) antenna. Nevertheless, all the elements used in their works were based on the traditional incremental finite element theory under the assumption of small rotations and deformations, which can not describe the system dynamic accurately. Mitsugi et al. [5] simulated the deployment process of a modular mesh antenna reflector, and the numerical results were also verified by a deployment experiment on a 4.8 m aperture reflector model.
Among different kinds of deployable antenna reflectors, the ring truss mesh reflector is one of the most widely used space antenna reflector due to its high packaging efficiency, light mass and high surface accuracy [6]. The AstroMesh reflector [6] developed by the Northrop Grumman Corporation has become one of the most advanced and reliable mesh reflector. However, the cable-net dynamics can not be accurately predicted by using the traditional incremental finite element because the cable-net will undergo large deformation and displacement in the reflector deployment process. The Absolute Nodal Coordinate Formulation (ANCF) originally proposed by Shabana [7], which can accurately deal with the dynamics of flexible multibody systems subjected to both large rotation and large deformation, has been considered as a benchmark in flexible multibody dynamics development. Liu et al. [8] proposed a composite laminated plate element of ANCF to simulate the deployment process of a SAR antenna. Recently, Peng et al. [9] studied the deployment dynamics of a flexible AstroMesh reflector ring truss. The ANCF and the Arbitrary Lagrangian-Euler description were used to establish a variable-length cable element. The non-synchronous deployment phenomenon between different parallelogram mechanisms of the reflector ring truss was successfully captured. However, the reflector cable-net was not considered in their dynamics model.

One of the most cumbersome computational tasks of the dynamic analysis for the flexible multibody system via the ANCF is to evaluate the element elastic force. Thus, efficient computational algorithms based on ANCF have attracted many researchers’ attention. Liu et al. [8] proposed an efficient formulation for evaluating the nonlinear elastic forces and the Jacobian of elastic forces via continuum mechanics theory. For the simulation of the long time reflector deployment process, the parallelization algorithm has to be developed to solve the extremely large set of equations. Liu et al. [8] embedded the OpenMP directives into the generalized-alpha algorithm and simulated the deployment process of a SAR antenna. The domain decomposition method is based on the divide-and-conquer concept, which partitions a large-scale system into a group of smaller systems. An application of the domain decomposition method is the substructure analysis [10], in which the total structure is considered to be an assemblage of many substructures (or subdomains). Then, the parallel solution of the equilibrium equations for these substructures can be performed. In field of structural mechanics, after domain decomposition, the static condensation method [11] is often further employed to eliminate the Degrees Of Freedom (DOFs) of the substructure internal nodes. However, after performing domain decomposition many additional constraints have to be added into the derived equations by using Lagrange multiplier method or penalty method, which will increase computational cost [12].

The objective of this study is to propose an efficient parallel algorithm to solve the huge sets of dynamics equations for deployment dynamics of mesh reflector with complex flexible cable-nets. The remainder of the paper is organized as follows. In Section 2, the equations of motion for an AstroMesh reflector are established. In Section 3, the domain decomposition approach and the static condensation method are introduced to solve the equation of motion. A recursive formulation is proposed in order to reduce the order of the simultaneous linear equations generated in Newton-Raphson iteration process. Finally, in Section 4, the deployment dynamics of a large-scale antenna system is investigated based on the proposed computational strategy. In Section 5, the main conclusions are drawn and recommendations for future studies are also outlined.
2 EQUATIONS OF MOTION FOR AN ASTROMESH REFLECTOR

As shown in Figure 1(a), an AstroMesh reflector is mainly composed of two lightweight paraboloidal cable-nets and a deployable ring truss, where the ring truss is constituted by several identical parallelogram mechanisms [6]. Each of parallelogram mechanism is composed of two longeron trusses, two batten trusses and two diagonal trusses as shown in Figure 1(b). The metallic mesh is attached to the backside of the front cable-net and is used for the antenna RF (radio frequency) reflective surface. The rear cable-net is the auxiliary cable-net to maintain the accuracy of the reflective surface.

![Figure 1](image)

\((a)\) The whole reflector under constraints \hspace{1cm} \((b)\) A parallelogram mechanism

The constraints of the reflector system include spherical joints, cylindrical joints, sliding hinges and gear joints. The longeron, batten and diagonal trusses are connected by the cylindrical joints at the truss joints where five trusses meet. The gear joints where only three bars are connected are used to maintain deployment synchronization between neighboring parallelogram mechanisms [6]. However, it has been clearly observed in the ground test that all parallelogram mechanisms can not deploy synchronously because of the truss deformation, the transmission errors of the gear joints and the decrease of the driving forces [6]. For the cable-net, the cables are connected with each other via spherical joints. The whole cable-net is connected with the ring truss also via spherical joints. In this study, to simulate a similar reflector boundary condition in space, it is assumed that one batten of the ring truss is fixed in space. As the nodal coordinates of ANCF elements are defined in the global coordinate system, the kinetic joints mentioned above can be easily formulated. In this study, the longerons, battens and diagonals of the ring truss are all considered as flexible beams and described via the fully parameterized ANCF beam element with 24 generalized coordinates proposed by Shabana and Yakoub [13]. For the reflector cable-net shown in Figure 1(a), the deformation of the cable cross-section is not considered. Thus, the gradient deficient ANCF beam element with only 12 nodal coordinates, which is originally proposed by Gerstmayr and Shabana [14], is used to mesh the antenna reflector cable-net. Using the ANCF, the equations of motions for the whole flexible reflector system can be expressed as a set of differential algebraic equations with a constant mass matrix [15],

\[
\begin{align*}
\mathbf{M}\dot{\mathbf{q}} + \mathbf{F}(\mathbf{q}) + \mathbf{D}(\mathbf{q}, \dot{\mathbf{q}}) + P\Phi_q^T\lambda + P_z\Phi_q^T\Phi - Q(\mathbf{q}, \dot{\mathbf{q}}) &= 0, \\
P\Phi(\mathbf{q}, t) &= 0,
\end{align*}
\]

where \(\mathbf{M}\) is a constant mass matrix of the system, \(\mathbf{q}\) is the generalized coordinates of the whole flexible multibody system, \(\mathbf{F}(\mathbf{q})\) is the elastic force vector of flexible bodies which is a strong nonlinear function of nodal coordinates, \(\mathbf{D}(\mathbf{q}, \dot{\mathbf{q}})\) is the damping force vector, \(\Phi(\mathbf{q}, t)\) is the constraint vector of the system, \(\Phi_q\) is the derivative matrix of constraint vector with respect to the generalized coordinates \(\mathbf{q}\), \(\lambda\) is the Lagrange
multiplier vector, $Q(q, \dot{q})$ is the generalized external force vector, $P_1$ and $P_2$ are the scaling coefficients for the constraint equations[15].

3 PARALLEL COMPUTATIONAL STRATEGY FOR MULTIBODY DYNAMIC EQUATIONS

3.1 Implement of the static condensation method for multibody system dynamics

In this study, the generalized-alpha algorithm [16] is used so as to eliminate contributions from nonphysical, high-frequency modes and preserve the responses of low frequencies well. When solving Equation (1) by using the generalized-alpha algorithm the following system of linear equations must be solved in the Newton iteration procedure,

$$
\begin{bmatrix}
\hat{\beta}M + F_q + \hat{\gamma}D_q + P_1\Phi_q^T\Phi_q - Q_q \\
0
\end{bmatrix}
\begin{bmatrix}
\Delta q \\
\Delta \lambda
\end{bmatrix}
= 
\begin{bmatrix}
M\dot{q} + F + D + P_1\Phi_q^T\lambda + P_2\Phi_q^T\Phi - Q \\
P_2\Phi
\end{bmatrix},
$$

(2)

where $\hat{\beta}$ and $\hat{\gamma}$ are the algorithm parameters and can be determined by the parameter $\rho \in [0, 1]$ of the algorithm. $D_q$ and $Q_q$ are the Jacobian of the damping force vector and the generalized external force vector, respectively.

In order to improve the computational efficiency, a flexible multibody system is firstly partitioned into $n$ independent subsystems. The static condensation method is employed to eliminate the subsystem’s internal generalized coordinates and Lagrange multipliers. According to Equation (2) for the $k$th subsystem the linear algebraic equations generated in the Newton-Raphson iteration can be written as

$$
\begin{bmatrix}
K_k & 0 \\
0 & 0
\end{bmatrix}
\begin{bmatrix}
P_k\Lambda_k^T & P_k^T \\
P_k^T & 0
\end{bmatrix}
\begin{bmatrix}
\Delta q_k \\
\Delta \lambda_k
\end{bmatrix}
= 
\begin{bmatrix}
F'_k \\
F_k^e
\end{bmatrix},
$$

(3)

where

$$
\begin{align*}
K_k &= \hat{\beta}M_k + (F_k)_q + \hat{\gamma}(D_k)_q - (Q_k)_q, \\
\Lambda_k &= (\Phi_k)_q, \\
F'_k &= M_k\ddot{q}_k + F_k + D_k + P_k(\Phi_k)^T\lambda_k + P_2(\Phi_k)^T\Phi_k - Q_k, \\
F_k^e &= P_k\left(\Phi_k + \sum_{i \in U_i, j \in k} (\Phi_k)^Tq_i \Delta q_i\right).
\end{align*}
$$

(4)

In Equation (4), $F_k$ and $q_k$ denote the elastic force vector and the generalized coordinate vector for the $k$th subsystem, respectively, and $\lambda_k$ is the Lagrange multiplier associated with the constraint $\Phi_k$ vector. $M_k$, $D_k$, $Q_k$, $\dot{q}_k$ and $\ddot{q}_k$ are the mass matrix, damping forces, external forces, generalized velocities and accelerations of the $k$th subsystem, respectively. Equation (3) be rewritten as

$$
\begin{bmatrix}
\Theta^u_k & \Theta^b_k \\
\Theta^lu_k & \Theta^lb_k
\end{bmatrix}
\begin{bmatrix}
\Delta q_k \\
\Delta \lambda_k
\end{bmatrix}
= 
\begin{bmatrix}
F'_k \\
F_k^e
\end{bmatrix},
$$

(5)
where $\Delta \mathbf{q}_i = \begin{bmatrix} \Delta q_{i1}^i \\ \Delta q_{i2}^i \end{bmatrix}$ denotes the increment of the $i$th subsystem’s internal variables and 

$\Delta \mathbf{q}_k^b = \begin{bmatrix} \Delta q_{k1}^b \\ \Delta q_{k2}^b \end{bmatrix}$ denotes the increment of the boundary variables. With the partitioned form $\mathbf{K}_k = \begin{bmatrix} \mathbf{K}_k^{ii} & \mathbf{K}_k^{ib} \\ \mathbf{K}_k^{bi} & \mathbf{K}_k^{bb} \end{bmatrix}$, $\mathbf{F}_k^i = \begin{bmatrix} \mathbf{F}_k^{ii} \\ \mathbf{F}_k^{ib} \end{bmatrix}$, $\mathbf{F}_k^c = \begin{bmatrix} \mathbf{F}_k^{ci} \\ \mathbf{F}_k^{cb} \end{bmatrix}$, the right-hand side and the coefficient matrix of Equation (5) can be respectively expressed as

$$
\mathbf{F}_k^i = \begin{bmatrix} \mathbf{F}_k^{ii} \\ \mathbf{F}_k^{ib} \end{bmatrix}, \quad \mathbf{F}_k^c = \begin{bmatrix} \mathbf{F}_k^{ci} \\ \mathbf{F}_k^{cb} \end{bmatrix},
$$

$$
\Theta_k^{ib} = \begin{bmatrix} \mathbf{K}_k^{ib} 0 \\ 0 0 \end{bmatrix}, \quad \Theta_k^{bi} = \begin{bmatrix} \mathbf{K}_k^{bi} 0 \\ 0 0 \end{bmatrix}, \quad \Theta_k^{bb} = \begin{bmatrix} \mathbf{K}_k^{bb} 0 \\ 0 0 \end{bmatrix} + \begin{bmatrix} P_2 \mathbf{A}_k^{bb} \mathbf{A}_k^b & P_1 \mathbf{A}_k^{bb} \end{bmatrix}.
$$

(6)

According to Equation (5), the increment of the $i$th subsystem’s internal variables can be calculated by

$$
\Delta \mathbf{q}_i = \mathbf{F}_k^i + \Theta_k^{ib} \Delta \mathbf{q}_k^b + \Theta_k^{bi} \Delta \mathbf{q}_k^b.
$$

(7)

Thus, substituting Equation (7) back into Equation (5) the increment of the boundary variables can be evaluated by

$$
\begin{bmatrix} \mathbf{K}_k & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} P_2 \mathbf{A}_k^{bb} \mathbf{A}_k^b & P_1 \mathbf{A}_k^{bb} \\ P_1 \mathbf{A}_k^{bb} & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{q}_k^b \\ \Delta \mathbf{q}_k^b \end{bmatrix} = \begin{bmatrix} \mathbf{F}_k^i \\ \mathbf{F}_k^c \end{bmatrix}.
$$

(8)

where

$$
\begin{align*}
\mathbf{K}_k &= \mathbf{K}_k^{ii} - \mathbf{K}_k^{ib} \mathbf{R}_k^c \mathbf{K}_k^{ib} \\
\mathbf{F}_k^i &= \mathbf{F}_k^{ii} - \mathbf{K}_k^{bi} \mathbf{R}_k^c \mathbf{F}_k^{ci} - \mathbf{K}_k^{bi} \mathbf{R}_k^c \mathbf{F}_k^{ci} \\
\mathbf{F}_k^c &= \mathbf{F}_k^{cb} \\
\mathbf{R}_k &= \left(\Theta_k^{bi}\right)^{-1} = \begin{bmatrix} \mathbf{R}_k^{ci} & \mathbf{R}_k^{cb} \\ \mathbf{R}_k^{ci} & \mathbf{R}_k^{cb} \end{bmatrix}.
\end{align*}
$$

(9)

By assembling the condensed subsystems’ equilibrium equations described by Equation (8) the system equilibrium equations can be expressed by

$$
\begin{bmatrix} \mathbf{K}_1 & 0 & 0 & 0 & 0 \\ 0 & \mathbf{K}_2 & 0 & 0 & 0 \\ 0 & 0 & \cdots & 0 & \cdots \\ 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix} + \begin{bmatrix} P_2 \mathbf{A}_1^{bb} \mathbf{A}_1^b & P_1 \mathbf{A}_1^{bb} \\ P_1 \mathbf{A}_1^{bb} & 0 \end{bmatrix} \begin{bmatrix} \Delta \mathbf{q}_1^b \\ \Delta \mathbf{q}_2^b \\ \cdots \\ \Delta \mathbf{q}_n^b \\ \Delta \lambda^b \end{bmatrix} = \begin{bmatrix} \mathbf{F}_1^i \\ \mathbf{F}_2^i \\ \cdots \\ \mathbf{F}_n^i \\ \mathbf{F}_c^i \end{bmatrix}.
$$

(10)

where $\Delta \lambda^b$ is obtained by assembling the terms $\Delta \lambda^b_k$, $\lambda^b$ and $\mathbf{F}_c^i$ are also obtained by assembling the terms $\lambda^b_k$ and $\mathbf{F}_c^i$, respectively. From Equation (10) it can be found that the evaluation of the coefficient matrices ($\mathbf{K}_k$, $k=1, 2, \ldots, n$) for different subsystems
and the right-hand side terms \( \bar{F}_k, k=1, 2, \ldots, n \) in Equation (8) can be parallelized. Similarly, the calculation of the internal variables for different subsystems can also be parallelized by substituting the results of Equation (10) back into Equation (7). It can be also found that the number of subsystems will affect the computation efficiency for solving Equation (10).

### 3.2 Recursive strategy based on the multilevel decomposition approach

To further reduce the order of Equation (10) and improve computation efficiency, a recursive strategy based on the multilevel decomposition approach is proposed in this subsection. The finite element mesh grid of a partitioned system with \( n \) subsystems is considered as the first level grid. After static condensation, the retained boundary nodes of each subsystem are regarded as the second level grid. The second level grid is further partitioned into different independently subsystems. Each subsystem in the second level is composed of several subsystems in the first level grid. Thus, the stiffness matrix and generalized forces of the subsystem in the second level can be formed by the stiffness matrices and generalized forces of the corresponding subsystems in the first level. By a similar way, the third, fourth, \ldots, and the \( h \)th level grid can be established. In the last level grid, only very small number of boundary nodes will be preserved. Thus, the order of the equilibrium equations for the last level grid will be small enough to be efficiently solved.

**Figure 2.** The schematic view of the multilevel decomposition approach.

Figure 2 gives a schematic view of the multilevel decomposition approach mentioned above. It can be clearly seen from Figure 2(a) that the cable-net with 30 interconnected cables is partitioned into 30 independent subsystems (sub 1-sub 30), and each subsystem is meshed by 2 slope deficient ANCF elements. In the first level grid, the DOFs of each subsystem inner-nodes can be condensed. According to Equation (8), after static condensation the equilibrium equations of a subsystem in the first level can be casted as
where \( k \) denotes the subsystem number, and \( h \) denotes the grid level number. \( \mathbf{K}_k^{(h)} \) and \( \mathbf{F}_k^{(h)} \) can be considered as the stiffness matrix and the generalized force vector of the \( k^{th} \) condensed subsystem, respectively.

To further reduce computational cost, for \( k = 1, 2, \ldots, 30 \), the condensed system are further partitioned into independently four smaller subsystems (sub 1-sub 4) as shown in Figure 2(b). The element grid of the four subsystems is considered as the second level grid. For each of the four subsystems the stiffness matrices and generalized force vectors can be assembled. For instance, the stiffness matrices and generalized force vectors of the subsystem 1 (shown in Figure 2(b)) are assembled from the subsystems 1, 2, 3, 4, 5, 6 and 7 of the first level grid (after condensation). Therefore, the equilibrium equations of a subsystem in the second level grid (before condensation) can be expressed as

\[
\begin{bmatrix}
\mathbf{K}_k^{(h)} & \mathbf{0} \\
\mathbf{0} & \mathbf{0}
\end{bmatrix}
+ \begin{bmatrix}
P_2 \mathbf{A}_k^{bT} \mathbf{A}_k^{b} & P_1 \mathbf{A}_k^{bT} \\
P_1 \mathbf{A}_k^{b} & \mathbf{0}
\end{bmatrix}
\begin{bmatrix}
\Delta \mathbf{q}_k^{(h)} \\
\Delta \mathbf{\lambda}_k^{(h)}
\end{bmatrix}
= - \begin{bmatrix}
\mathbf{F}_k^{(h)} \\
\mathbf{0}
\end{bmatrix},
\tag{11}
\]

After condensation operations, the equilibrium equations for a subsystem in the second level grid can be written as the same form as Equation (11). Equation (11) will still not be directly assembled and solved, after condensation operations the element grid of the four subsystems is considered as the third level grid. As shown in Figure 2(c), the third level grid can be also considered as one subsystem. The equilibrium equations for the third level grid can be assembled by the four subsystems of the second level, which are similar to Equation (12).

Therefore, according to the above multilevel decomposition process, a series of multilevel formulations to evaluate the stiffness matrix and generalized force vector for different level grids can be obtained by

\[
\begin{bmatrix}
\mathbf{K}_k^{(h)} \\
\mathbf{0}
\end{bmatrix}
= \begin{bmatrix}
\mathbf{K}_k^{(h)} \\
\mathbf{0}
\end{bmatrix}
+ \begin{bmatrix}
P_2 \mathbf{A}_k^{bT} \mathbf{A}_k^{b} & P_1 \mathbf{A}_k^{bT} \\
P_1 \mathbf{A}_k^{b} & \mathbf{0}
\end{bmatrix}
\begin{bmatrix}
\Delta \mathbf{q}_k^{(h)} \\
\Delta \mathbf{\lambda}_k^{(h)}
\end{bmatrix}
\tag{13}
\]

where \( \mathbf{A}_j^{(h-1)} \) denotes the assembly over the \( n_k \) subsystems of the \((h-1)\)th level grid. As shown in Figure 2(d), once the multilevel procedure reaches the last level, the boundary generalized coordinates and Lagrange multipliers can be obtained by solving the reduced equivalent equilibrium equations Equation (11) for only 12 element nodes. After all the results for the last level grid were obtained, a back substitution process from the last level grid to the first level grid can be performed to calculate the results for other level grids. Finally, back substitution formulations to evaluate the increment of variables in different level grids can be written as
\[
\Delta \mathbf{q}^{(h-1)}_j = \left[ \left( \Delta \mathbf{q}^{(h)}_k \right)^\top \left( \Delta \mathbf{\lambda}^{(h)}_k \right)^\top \left( \Delta \mathbf{q}^{(h)}_i \right)^\top \left( \Delta \mathbf{\lambda}^{(h)}_i \right)^\top \right]_{jk},
\]
\[
\Delta \mathbf{q}^{(h-1)}_j = -\left( \left( \mathbf{\Theta}^{(h)}_j \right)^{-1} \left( \mathbf{\Theta}^{(h)}_j \Delta \mathbf{q}^{(h)}_j + \mathbf{F}_j \right) \right)^{h-1}, \quad j = 1, 2, \ldots, n_k. \tag{14}
\]

Thus, from above deduction process the procedures for the multilevel decomposition method can be generalized as: firstly, using Equation (13), the essential element information for different level grids, including the stiffness matrices and generalized force vectors can be recursively calculated and stored in computer memory. Then, the generalized coordinates and Lagrange multipliers in the last level grid can be directly obtained by solving Equations (11) and (14). Finally, the generalized coordinates and the Lagrange multipliers of the other level grids can be obtained by using the back substitution formulations described by Equation (14).

4 CASE STUDIES AND DISCUSSIONS

As shown in Figure 3(a), a modeling and analysis of a ring truss AstroMesh reflector is studied. The antenna aperture and the height of the batten are assumed as 20 m and 2.5 m, respectively. The lengths of the longerons can be calculated according to the reflector aperture, and it is 3.473 m. Figure 3(b) shows the fully deployed reflector ring truss, which are assembled by using the fixed joints, cylindrical joints, sliding hinges and gear joints. In this study the ring truss consists of eighteen parallelogram mechanisms shown in Figure 3(b). The batten \( O_1O_2 \) of the ring truss is fixed in space to simulate a similar reflector boundary condition in space. The spatial cable-net is composed of 648 front cables, 211 tension ties and 648 rear cables.

As shown in Figure 3(b), the deployment angle \( \phi \) of the parallelogram mechanism is selected as a parameter to determine the reflector deployment state, and if the deployment angle is approaching \( \pi/2 \), the corresponding parallelogram mechanism will be locked. The parallelogram mechanism will become a parallelogram structure. The cross sections of the battens, longerons and diagonals are all annulus shape with inner radius 0.014 m and outer radius 0.015 m. For the ring truss the Young’s modulus of the material is assumed to be \( 1.6 \times 10^{11} \) Pa, the Poisson’s ratio is assumed to be 0.3 and the material density is 1600 kg/m\(^3\). For the cable-net the Young’s modulus of the material is assumed to be \( 0.5 \times 10^{11} \) Pa, the Poisson’s ratio is assumed to be zero and the material density is 1800 kg/m\(^3\). The radius of the cable circular cross-section is set as 0.9 mm. In addition, the lumped mass for the joints connecting five trusses is set as 0.5 kg while the
lumped mass for the joints connecting only three trusses is set as 0.3 kg, as shown in Figure 3(a). The integration time step is $1 \times 10^{-3}$ s and the tolerate error is set as $1 \times 10^{-6}$. In this study, numerical simulations were performed on a workstation with twelve 3.33 GHz cores and a 96 GB RAM. For parallelization, twelve threads are used.

In this example, each truss and each cable are meshed by 8 fully parameterized ANCF beam elements and 20 slope deficient ANCF cable elements described in Section 2, respectively. As mentioned in Section 2, the deployment of the system is achieved by applying the driving forces with their directions along the diagonal trusses, as shown in Figure 4. Here, the driving forces are symmetrical along the X-axis (Figure 3(b)) due to the symmetry of the ring truss. However, the magnitude of the driving forces along the diagonal truss of different parallelogram mechanisms will decrease due to the friction effects. Thus, in this study the decreasing forces are simply defined as

$$F_{i+1} = CF_i, \quad i = 1, 2, ..., 8 \tag{15}$$

where $C$ is the damping coefficient of the driving force, and is set as 0.729. $F_i$ is the driving force for the $i$th parallelogram mechanism, as shown in Figure 4.

In this example, each truss and each cable are meshed by 8 fully parameterized ANCF beam elements and 20 slope deficient ANCF cable elements described in Section 2, respectively. As mentioned in Section 2, the deployment of the system is achieved by applying the driving forces with their directions along the diagonal trusses, as shown in Figure 4. Here, the driving forces are symmetrical along the X-axis (Figure 3(b)) due to the symmetry of the ring truss. However, the magnitude of the driving forces along the diagonal truss of different parallelogram mechanisms will decrease due to the friction effects. Thus, in this study the decreasing forces are simply defined as

$$F_{i+1} = CF_i, \quad i = 1, 2, ..., 8 \tag{15}$$

where $C$ is the damping coefficient of the driving force, and is set as 0.729. $F_i$ is the driving force for the $i$th parallelogram mechanism, as shown in Figure 4.

$$F_i = \begin{cases} a_5 \times t^5 + a_4 \times t^4 + a_3 \times t^3 + a_2 \times t^2 + a_1 \times t + a_0, & 0 < t \leq 212 \\ b_1 + 0.2 \times (t-212), & 212 < t \leq 408, \\ b_2 + 0.5 \times (t-408), & 408 < t \leq 500 \end{cases} \tag{16}$$

where $a_5 = -4.1 \times 10^{-10}$, $a_4 = 2.6 \times 10^{-7}$, $a_3 = -5.8 \times 10^{-5}$, $a_2 = 4.7 \times 10^{-3}$, $a_1 = 8.5 \times 10^{-3}$, $a_0 = 2.5$, $b_1 = 1.7 \times 10^1$, $b_2 = 5.6 \times 10^1$. Moreover, the maximum static friction force along the direction opposite to the direction of the driving forces is assumed to be 5 N.

In order to investigate the influence of the ring truss flexibility on deployment angle results, the deployment dynamics of a reflector with truss material Young’s modulus $E=8.0 \times 10^{12}$ Pa is also studied. The result curves for the deployment angles $\theta_i, i=1, 2, ..., 9$, are given in Figure 5. Obviously, it can be seen from Figure 5 that all the result curves almost coincide with each other. It can also be observed that the sharp result changes happened in the period from 105 s to 120 s. The main reason for this phenomenon is that in that period the cable-net begin to be stretched by the tension forces obtained from form-finding process. The reflector will continue to deploy if the driving force is large enough to balance the cable-net tension forces after 250 s. In addition, from Figure 5 the non-synchronous deployment phenomenon for the relative flexible system with ring truss material Young’s modulus $E=1.6 \times 10^{11}$ Pa can be obviously observed.
To study the deformation behaviors of the ring truss, Figure 6(a) firstly gives the schematic view of the ring truss transverse deflection, which is the maximum distance between the points on the batten and the line $i$-$j$. Figure 6(b) gives the transverse deflections of all the battens. Because the reflector is symmetrical with respect to $X$-axis shown in Figure 3(b), only the results for nine battens are shown in Figure 6(b). It can be seen from Figure 6(b) that the transverse deflections of the first, second, third, fourth and fifth battens are a bit larger than those of other battens. It can also be found from this figure that after a continuous increasing period the deflection values of the first, second, third, fourth and fifth battens sharply decreased at some specific moments. The reason is that at these specific moments the deployment angles of the corresponding parallelogram mechanisms are close to $\pi/2$, these battens will be locked, which lead to a sharp increasing of the these parallelogram mechanisms’ equivalent stiffness.

Finally, 6 dynamic configurations of the system with respect to 6 specific simulation moments are given in Figure 7. From Figure 7, the non-synchronous deployment phenomenon during the deployment of the ring truss is obviously observed, which is consistent with the results in ground experiment test [9]. The total cost CPU time for the deployment process simulation is about 78 h.
5 CONCLUSIONS

In this study, the finite element model of a flexible large-scale ring truss deployable antenna is established based on ANCF. The domain decomposition method is introduced to divide a flexible multibody system into several independent subsystems and then the internal generalized coordinates and the Lagrange multipliers of each subsystem are eliminated by using the static condensation method. To further improve the computational efficiency, the finite element mesh grid assembled by the retained boundary variables of subsystems can be further partitioned based on the multilevel decomposition approach. Besides, the essential element information for different level grids, including the stiffness matrices and generalized force vectors can be recursively calculated by using the proposed recursive formulation. Thus, the DOFs of the last level grid are significantly reduced compared with those of the original finite element model. Based on this parallel computational methodology, the deployment dynamics of a complex AstroMesh reflector is carefully investigated including the deployment angles and the elastic deformation of the ring truss. The numerical results indicate that the elastic deformation of flexible parts will cause the non-synchronization phenomenon during the deployment of the antenna. The obtained results can be referred for the design, prediction and control of the large-scale space deployable structures. In the further research, the numerical results will be validated by ground experiment tests.

REFERENCES


Development of efficient flexible multibody techniques for rotordynamics systems including rotors and supporting structures

S. Rossin*, A. Frilli#, E. Meli#, G. Pallini#, S. Panconi#, A. Rindi#

* General Electrics Nuovo Pignone Oil&Gas Via di Santa Marta 3, 50139, Florence, Italy stefano.rossin@ge.com

# Department of Industrial Engineering University of Florence Via Matteucci 2, 50127, Florence, Italy amedeo.frilli@unifi.it enrico.meli@unifi.it giovanni.pallini@unifi.it simone.panconi@unifi.it andrea.rindi@unifi.it

ABSTRACT

In the study of the rotor dynamic performances of a rotating machine, it is possible to divide the plant in four main components: rotor, stator, bearings and supporting structure. The current trend for rotating machinery considers the reduction of weight an important problem. As a consequence of the complex behavior the last generation plants, characterized by low weights and flexible support structures, the separation margin between rotor critical speeds and operative conditions is quite difficult to be reached. The presence of flexible supports can generate dangerous coupling effects among the different bearings. Therefore the model of a complete rotor dynamical system must consider all the phenomena coming from the coupling of its mechanical components [1]. The aim of this work is to demonstrate the importance of taking into account the support structure since the preliminary project phases. To this end, this study investigates a problem of enhanced vibration level on a five stage MCL rotor, part of a compression system made up of a gas turbine driver with two centrifugal compressors and one gearbox. Each rotating machine was installed on a separate baseplate steel structure and the individual rotors were connected by elastic coupling. This research work has been developed cooperating with Nuovo Pignone General Electric S.p.a., that has provided the technical and experimental data required for the model validation.

Keywords: Rotating machines, Supporting structures, Bearings, Dynamical bearing interaction.

1 INTRODUCTION

A rotordynamics machinery is a very complex component composed of many elements which are linked each other. In particular this research work focus on the interaction of the following components:

1. rotor;
2. bearings;
3. support structure.

Because of the need of high efficiency, the turbomachinery development trend brings towards more and more compact systems characterized by reduced weight. Consequently, the structures must necessarily become more lighter and flexible. The introduction of system components that can not be considered absolutely rigid in the operative range, leads to the need of a deeper study of the dynamic behaviour of rotating machines to guarantee safe operating conditions. For example the support structures and the anchoring system (visible in Figure 1) for off-shore installations is isostatic because of deck flexibility. This is obtained through the use of hinge and spherical joints,
which allow the deck deflection. At the same time, because of support structures and anchoring system, more complex shape modes appear.

According to ISO and API norms, to estimate the effect of the supporting structure on the rotor dynamic behaviour (as visible in Figure 2), one can compare the stiffness of supporting structure at a specific location, usually one of bearings seat, with a limit defined upon the maximum expected vibration under maximum unbalance load.

As visible in Figure 2, where the comparison among the dynamic stiffnesses measured at the drive end and not drive end side and the vibration limit is reported, the lack of stiffness corresponds to a possible resonance of the compressor due to supporting structure elasticity.

The dynamics of the support structure couples the degrees of freedom of the different bearings of the machine and the different DOFs of the same bearing (Figure 3). Consequently, the effect of this coupling on the shaft line and the role played by the supporting structures have to be accurately studied.

The problem now is how to introduce the dynamic behavior of the support system in order to obtain an accurate and efficient dynamical model of the whole system [2]. The transfer function (see Equation 1) [3] of the baseplate may be divided into different parts: direct and cross-talking terms as visible in Figure 4 [4]:

\[
\alpha_{bas} = \begin{bmatrix} \alpha_{11} & \alpha_{12} \\ \alpha_{21} & \alpha_{22} \end{bmatrix}. \tag{1}
\]
Figure 3. Rotor mode shape due to the supporting structures

It comprises four sub matrices: direct terms, $\alpha_{11}$ and $\alpha_{22}$, and cross-talking terms, $\alpha_{12}$ and $\alpha_{21}$. The direct terms can be thought of as springs in series to the bearings, while the cross-talking terms represent functions coupling the dynamic behavior of the bearings. The second terms are characteristic of all the off-shore installations where the bearings behaviors on the same shaft line can not be considered uncoupled because of the foundation presence [5].

Figure 4. Foundation transfer function

To study the effect of the support structure on the dynamic behaviour of the rotor, three steps are necessary:

1. to analyze the whole system without simplification;
2. to build a reduced model to improve the numerical efficiency (based on Component modal analysis techniques);
3. to build a model to separate the influence of the different bearings.

The full model is modeled using the ANSYS software. In particular, the rotor is modeled through 3D beam elements, with a circular section. Each node, has 6 DOFs (3 translations and three rotations). The rotor model is composed of the sum of two contributes: the first one, is the contribute of the shaft, characterized by the properties of steel while the second one is the contribute of added masses, arranged along the shaft axis. The added masses represent the assembled elements on the shaft as impellers and seals. The baseplate includes the case discretization which is supposed to be much more rigid than the other components and so is represented by concentrated properties of mass and inertia applied to its own center of gravity. The foundation is discretized with shell elements and the stiffness of the anchorage springs to the ground is extracted from dynamic stiffness simulations to match the results of hammer tests on the supporting pedestals. The bearings has been modeled by stiffness and damping matrices which depend on the angular velocity of the rotor. These elements connect the rotor to the foundation.
To improve the efficiency of the model, because of the large number of DOFs, a reduced model has been built, using component mode synthesis techniques (CMS). In particular the original supporting structure transfer function $\alpha_{\text{bas}}$ has been reduced obtaining a simplified transfer function $\alpha_{\text{red}}^{\text{bas}}$.

To evaluate the influence of the different bearings, a diagonal reduced model has been built. This way, the supporting structure model has been further reduced to the following transfer function:

$$\alpha_{\text{bas}}^{\text{dia}} = \begin{bmatrix} \alpha_{11} & 0 \\ 0 & \alpha_{22} \end{bmatrix}. \quad (2)$$

Considering a diagonal reduced matrix, means to neglect the crosstalk terms and, consequently, the coupling between the bearings.

The considered system (the study case) is an application for onshore installation, whose experimental characteristic are provided by Nuovo Pignone GE.

2 GENERAL ARCHITECTURE

A scheme of the general architecture of the model is represented in Figure 5. The whole system is composed of three components, the rotor, bearings and the support structure[6]. The bearings, that allow the relative motion between rotor and support structure, are characterized by stiffness and damping matrices and affect the rotor dynamic behaviour. The bearing reactions, caused by the rotor motion, are applied to the support structure which in turn reacts according to its dynamics. The components of the whole system are extremely coupled to each other and the dynamic behaviour of a single part can not be studied singularly. In Figure 5 one can see the various steps of the analysis:

1. first step: the complete model of rotor and basament is developed (transfer function $\alpha_{\text{bas}}$);
2. second step: the support structure model is reduced through CMS technique (obtaining $\alpha_{\text{red}}^{\text{bas}}$);
3. third step: starting from the second step, a diagonal reduced transfer function ($\alpha_{\text{bas}}^{\text{dia}}$) of the basement is extracted. This way, the transfer function is composed of only two sub matrices. Each sub matrices is full and composed of 6x6 elements, as the DOFs of master nodes are 6.

3 THE COMPLETE MODEL

3.1 The Rotor Model

The rotor, as visible in Figure 5 is described by the following equation:
\[ M_{\text{rot}} \ddot{u}_{\text{rot}} + (C_{\text{rot}} + \Omega G_{\text{rot}}) \dot{u}_{\text{rot}} + K_{\text{rot}} u_{\text{rot}} = F_{\text{rot}} + F_{\text{bear}} \]  

where \( M_{\text{rot}}, C_{\text{rot}}, G_{\text{rot}}, \) and \( K_{\text{rot}} \) are the damping, gyroscopic effects and stiffness matrices (including also the centrifugal effects) respectively; \( u_{\text{rot}}, F_{\text{rot}} \) and \( F_{\text{bear}} \) are the displacement vector, the external force vectors and the bearings reactions forces. If the external forces acting on the rotor are harmonic [7], Equation (3) can be expressed in the following form:

\[ \alpha_{\text{rot}}^{-1}(\omega) u_{0,\text{rot}} = F_{0,\text{rot}} + F_{0,\text{bear}} \]  

where \( \omega \) is the excitation frequency of the forces.

3.2 Bearings

When foundation and rotor are assembled, the connection elements between these components are the bearings. These elements have stiffness and damping properties depending on rotor angular velocity[8]. For example, if the revolution axis of the rotor is the \( z \) axis, the bearing will have stiffness and damping in \( x \) and \( y \) directions as visible in Figure 6: The bearing dynamics is represented by the following equation:

\[
F_{\text{bear}} = \begin{bmatrix}
K_{xx}(\omega) & K_{xy}(\omega) \\
K_{yx}(\omega) & K_{yy}(\omega)
\end{bmatrix} u_{\text{bear}} + \begin{bmatrix}
C_{xx}(\omega) & C_{xy}(\omega) \\
C_{yx}(\omega) & C_{yy}(\omega)
\end{bmatrix} \dot{u}_{\text{bear}}
\]  

where \( u_{\text{bear}} \) is the difference between the rotor and support structure displacements.

These stiffness and damping characteristics influence the dynamic behavior both of the rotor and of the foundation. Analyzing in detail the matrices representing the bearings, four elements can be identified. The direct stiffness coefficients \( K_{xx} \) and \( K_{yy} \) produce a radial force directed along with the rotor deflection vector. If the coefficients are negative, the forces have reverse directions. The direct damping coefficients \( C_{xx} \) and \( C_{yy} \) produce a tangential force against the whirl velocity.

The cross-coupled stiffness coefficients \( K_{yz} \) and \( K_{xy} \) produce a tangential force normal to the deflection vector while cross-coupled damping coefficients \( C_{yz} \) and \( C_{yx} \) are responsible for the radial force co-linear with the deflection vector and their direction depends on their algebraic sign.
3.3 Support Structure

The foundation, visible in Figure 7, has been discretized by shell elements to represent the steel structure and the rigid connections (which are the links between rotor and baseplate) and to describe the casing modelling. Describing the foundation through shell elements means to extract the average surface of support structure components. This way the model is computationally less heavy but an approximation is made on the third dimension (it is a good compromise between accuracy and efficiency when the thickness is negligible if compared to the other dimensions). The equation of motion of the baseplate is described as follows in the time domain:

\[ M_{bas} \ddot{u}_{bas} + C_{bas} \dot{u}_{rot} + K_{bas} u_{bas} = F_{bas} + F_{bear} \]  

(6)

and in the frequency domain:

\[ \alpha_{bas}^{-1} (\omega) u_{0, bas} = F_{0, bas} + F_{0, bear}. \]  

(7)

In Equation (6), \( M_{bas}, C_{bas} \) and \( K_{bas} \) are the mass, damp and stiffness matrices of the baseplate, while \( F_{bas} \) and \( F_{bear} \) are the external forces acting on the baseplate and the bearing reactions. The vector \( u_{bas} \) contains all the degrees of freedom of the foundation.

![Figure 7. Baseplate of the test case](image)

3.4 The Complete Plant

The complete model is composed of the elements listed above. To describe the equations of motion, it is necessary to define a new vector (\( \bar{u} \)) of the generalized coordinates including a rearrangement of all the degrees of freedom of the components. The dynamic equations of the whole system are described as follows:

\[ \ddot{\bar{u}} + (\tilde{C} + \Omega \tilde{G}) \dot{\bar{u}} + \tilde{K} \bar{u} = F_{ext} \]  

(8)

where \( \tilde{M} \) represents the mass of the assembled system, \( \tilde{G} \) is the gyroscopic effects matrix related to the rotating parts, \( \tilde{C} \) and \( \tilde{K} \) are the damping and stiffness matrices including the bearings contribute. Hypothesizing that the external forces \( F_{ext} \) are harmonic (\( F_{0, ext} e^{i\omega t} \)), the displacements will be: \( \bar{u}_0 e^{i\omega t} \). Substituting the expressions of displacements and forces in Equation (8), the dynamic equation of the complete model becomes:

\[ \alpha_{tot}^{-1} (\omega) u_0 = F_{0, ext} \]  

(9)

4 THE REDUCED MODELS

The baseplate is the heaviest computational component because of the large number of elements used to discretize it. The baseplate is then reduced to a simpler transfer function by CMS technique.
in particular Craig and Bampton method. To reduce the basement, it is necessary to choose nodes, called master nodes, representing the link between the foundation and other components. The node chosen to represent the basement are the nodes on the bearing seat as visible in Figure 7. This way, the matrix dimension has been reduced from \( m + n \), sum of degrees of freedom of master \( (m) \) and slave coordinates \( (n) \), to \( m + k \), where \( k \) are the vibration modes chosen to represent the foundation. Each master node has six degrees of freedom and the modal base chosen includes all the frequencies characterizing the operative range of the machinery, up to 10000 rpm. The support structure reduced model is represented by following equation:

\[
\alpha_{\text{red}}^{-1}(\omega)\mathbf{u}_{\text{bas}} = F_{0,\text{ext}}
\]

In this way the model dimension has been reduced from 537132 degree of freedom in the full model to 32 degree of freedoms in the reduced model, without considering the rotor models. Furthermore, to highlight the interaction between different bearings of the same machine, the support structure transfer matrix \( \alpha_{\text{red}}^{-1} \) has been simplified to obtain a diagonal matrix. The equation governing the support structure is described as follows:

\[
\alpha_{\text{dia}}^{-1}(\omega)\mathbf{u}_{\text{bas}} = F_{0,\text{ext}}
\]

Considering a diagonal reduced transfer function (Equation (11)) means that each bearing is connected to a different support structure and the coupling effects are neglected. The rotor and the bearing models are the same previously described.

5 TEST CASE

The study case is composed of a gas turbine driver, two centrifugal compressors and one gearbox in one single shaft line. Each rotating machine is installed on a separate baseplate structure and the rotors are connected by elastic coupling. The choose of analyzing this test case, is due to the strong effects of the support structure on the dynamic behaviour of the centrifugal compressor, as it will be explained in the following. During the experimental tests, high vibrations have been observed on the centrifugal compressor at the proximity probes of the bearings. The vibrations were quite high and compromised the right start-up of the machinery. Several accelerometer probes have been arranged on the multi-stage compressor. Particularly the probes have been placed at the Drive End (DE) and Not Drive End (NDE), at the bearings and on the baseplate structure as visible in Fig. 8.

![Figure 8. Accelerometers probes arrangement](image)

The acquired data are the vertical and horizontal displacements and, consequently, amplitude and phase. In particular the rotor displacements, measured at the bearings (NDE), are visible in Figure 9. The rotor shows a critical speed at around 3300 rpm. The acquired data on the support structure are visible in Fig.10, in particular the displacements along the vertical axis at DE side are showed. The acquisition point on the support structure for the experimental data is showed in Fig.8, identified by S_DE-DX label.
As visible in Figure 11, where the undamped critical speed map is showed, the rotor (without the influence of the support structure) has a critical speed around 3800 rpm. This means that the support structure presence shifts the rotor critical speed from 3800 rpm to 3300 rpm.

Concerning the baseplate, the frequencies computed through the finite element models (without considering the rotor) have been compared to the ones measured on the machine at NDE. In Figure 12 the waterfall plot extracted from the run-up test of the whole machine recorded on the supporting structure is represented. In the Table 1 the support structure critical speeds (both the measured ones, see Figure 12, and the simulated ones obtained analyzing only the foundation) are listed.

As visible in Table 1, there are some differences between the simulated frequencies of the single component and the ones measured on the machine. This is due both to the approximation of the model and to the effects of support structure on the system.

6 VALIDATION OF THE MODEL

To validate the model, an imbalance response analysis has been performed. In Figure 13(a), it is possible to see the displacements curve at the bearing station (NDE). In particular a comparison between the first step (complete model) and the second step (reduced model) is showed. Both models are able to described the behaviour of the system. The resonance peak around 3300 rpm is present in both the curves. The imbalance response analysis is in good agreement with the experimental data reported in Figure 9, both in terms of frequencies and in terms of amplitude. To understand the influence of the coupling effect between the different bearings due to the base-plate, the diagonal reduced model has been analyzed as well. The imbalance response analysis is reported in Figure 13(b). This time the resonance peak has been shifted to around 3800 rpm, while the amplitude turns out to be a little different if compared to Figure 13(b). This is reasonable.
because, neglecting the cross-talk terms, the support structure do not cause the bearings coupling. The support structure velocities measured at the compressor foot, DE side, are reported in Fig.14.

Comparing the simulated results with the experimental ones, visible in Fig.10, the finite element model is able to reproduce the support structure behaviour. The velocities, obtained from simulations, are in agree with the experimental data, both in amplitude (0.05 in/s 0-pk) and in the speed range in which they occur (3000 – 4000 rpm). Furthermore, in Table 2(a) and in Table 2(b) the simulated frequencies related to the support structure and to the rotor obtained analyzing the whole system are listed.

Observing the two tables, also in this case one can note that the results, obtained with diagonal reduced model (III step) are quite different from the ones obtained through the complete (I step) and reduced (II step) models. This is due the crosstalk terms which couple the bearings behaviours, that have been neglected. On the contrary, the frequencies produced by the whole model (I and

Table 1. Support structure frequencies

<table>
<thead>
<tr>
<th>Experimental freq.</th>
<th>Simulated freq.</th>
</tr>
</thead>
<tbody>
<tr>
<td>29 Hz</td>
<td>29 Hz</td>
</tr>
<tr>
<td>41 Hz</td>
<td>41 Hz</td>
</tr>
<tr>
<td>50 Hz</td>
<td>49 Hz</td>
</tr>
<tr>
<td>70 Hz</td>
<td>68 Hz</td>
</tr>
<tr>
<td>86 Hz</td>
<td>90 Hz</td>
</tr>
</tbody>
</table>
Figure 14. Support velocities at DE

(a) Related to the support structure

<table>
<thead>
<tr>
<th>I STEP</th>
<th>II STEP</th>
<th>III STEP</th>
</tr>
</thead>
<tbody>
<tr>
<td>29 Hz</td>
<td>29 Hz</td>
<td>25 Hz</td>
</tr>
<tr>
<td>41 Hz</td>
<td>41 Hz</td>
<td>36 Hz</td>
</tr>
<tr>
<td>50 Hz</td>
<td>50 Hz</td>
<td>46 Hz</td>
</tr>
<tr>
<td>69 Hz</td>
<td>69 Hz</td>
<td>65 Hz</td>
</tr>
<tr>
<td>88 Hz</td>
<td>88 Hz</td>
<td>84 Hz</td>
</tr>
</tbody>
</table>

(b) Related to the rotor

<table>
<thead>
<tr>
<th>I STEP</th>
<th>II STEP</th>
<th>III STEP</th>
</tr>
</thead>
<tbody>
<tr>
<td>55 Hz</td>
<td>55 Hz</td>
<td>63 Hz</td>
</tr>
</tbody>
</table>

Table 2. Frequencies of whole system

II step) well match the experimental data reported in Figure 9 and in Figure 12 (better than the uncoupled analysis).

7 CONCLUSIONS

In this work the authors investigated the role played by elastic supporting structures in the whole rotor-bearings-baseplate system. To this end, the authors developed a model able to accurately reproduce the dynamical behaviour of the complete system and, at the same time, to reach a good compromise between accuracy and numerical efficiency. The presented approach and the developed model have been tested through experimental data provided by Nuovo Pignone GE and related to a critical rotating machine-baseplate system. The validation highlighted the accuracy of the model and the results are encouraging. The future developments of the research activity will consist in further validations of the approach through new experimental data and in the application of the proposed strategy to more complex plants involving more rotating machines interacting to each other.

REFERENCES


Bibliography:


Dynamic Analysis of the Rotating Thin-walled Structure via the Absolute Nodal Coordinate Formulation

Chunzhang ZHAO#*, Haidong YU*, Hao WANG#, Yong ZHAO*

# Key Laboratory of Mechanical System and Vibration
Shanghai Jiao Tong University
Shanghai, 200240, P.R. China
[zez, wanghao]@sjtu.edu.cn

* Key Laboratory of Digital Manufacture for Thin-walled Structures
Shanghai Jiao Tong University
Shanghai, 200240, P.R. China
[hdyu, zhaoyong]@sjtu.edu.cn

ABSTRACT

Thin-walled structures are widely used in modern engineering. Large deformations are easy to occur in these structures due to their geometric features, especially when they experience high-speed rotation, which affect the dynamic behavior and stability of the system. So it is necessary to establish a precise flexible dynamic model of the thin-walled structure to study its dynamic behavior during high-speed rotation. In this study, the absolute nodal coordinate formulation (ANCF) method is used to establish the dynamic model and to study the modal characteristic of a thin-wall structure which rotates around the axis across its center with a fixed angular velocity. A four-node plate element of ANCF with 48DOFs is employed to discrete the thin-walled structure. The vector of the element elastic forces and the stiffness matrix are derived based on continuum mechanics. The rotating thin-walled structure is equivalently treated as a nonrotating one subject to an external force and the analytical expressions of the Jacobian formulation of elastic forces are derived. Static deformation tests of a cantilever plate and the dynamic tests of a pendulum plate are used to validate the formulations for the plate element of ANCF. The modal characteristics of a plate rotating around the axis across center are investigated and the effect of the angular velocities on the modal characteristics of the plate is studied.

Keywords: ANCF, Rotating plate, Modal Analysis, Plate element.

1 INTRODUCTION

Thin-walled structures are widely used in modern engineering, such as the solar energy panels, the vehicle bodies, and et al. Large deformations are easy to occur in these structures due to their geometric features, especially when they experience high-speed rotation. The flexible deformation of the thin-walled structures will affect the kinematic accuracy and the dynamic behavior of the whole mechanisms, especially the dynamic stability of the system. For example, radial rotating motion of a cantilever plate, induces centrifugal inertia force which results in the stretching of structure and effectively increases its bending stiffness. So it is necessary to establish a precise flexible dynamic model of the thin-walled structure to study its flexible deformation and dynamic behaviors during high-speed rotation.

In the early studies, rotating flexible structures were often idealized as rotating beams for that such idealization has provided accurate dynamic characteristics for most rotating flexible structures. Based on the Rayleigh energy theorem, Southwell and Gough [1] firstly proposed an analytical method to calculate the natural frequencies of a rotating beam. This equation which is known as Southwell equation is still widely used today for its simple form. To obtain more accurate natural frequencies, Schilhansl [2] derived a partial differential equation of a rotating beam and applied Ritz method to it. In this way, more accurate coefficients for thee Southwell equation could be obtained. Chung and Yoo [3] presented a finite element analysis for a rotating cantilever beam and computed the natural frequencies, time responses, and distributions of the deformations and stresses. Although many rotating structures can be successfully idealized as beams, they are not accurate enough for modeling structures which behave like plates rather beams. Rotating plates are obviously better models for structures such as solar panels and turbo-machine blades [4]. Dokainish and Rawtani [5] used a finite element techniques to determine the modal characteristics of rotating cantilever plates mounted on the periphery of a rotating rigid hub. They analyzed the relationship between the natural frequencies and the system parameters. Ramamurti and Kielb [6] used a similar approach in order to determine the modal properties of twisted rotating plates. Yoo et al. [7-8] derived a set of rotating cantilever plates
and investigated the effects of the dimensionless parameters on the modal characteristics of rotating cantilever plates.

The absolute nodal coordinate formulation (ANCF), which was proposed by Shabana [9] in 1996, has been successfully developed and is one of the research focus in multibody system dynamics areas recent years. In the ANCF, slopes are used instead of infinitesimal rotations to describe element configuration. All the coordinates are defined in the global coordinate system. The hypothesis of the small deformation and the infinitesimal rotation in other three methods are released. Therefore, this method has a high precision comparing with other methods such as the kineto-elasto dynamic method, the floating frame of reference method and the finite element method when used to describe deformable bodies experience large deformations [10]. Many scholars have contributed to the modeling of ANCF plate element. Shabana and Christensen [11] proposed the first ANCF plate element. The element has 48DOFs and is rigidly connected to each other at nodes, so only the displacements can be continuous at the nodes. A full parametrized rectangular plate/shell element with 48DOFs was proposed by Mikkola and Shabana [12]. Nodal positions and three directional slopes were introduced as the nodal coordinates. They also used a continuum mechanics method to describe the deformation of the element. Dmitrochenko and Pogorelov [13] proposed a way to generate several thin plate elements of ANCF to overcome the numerical difficulties occurred when the multibody system includes very thin and stiff plates. By eliminating the nodal coordinates of the position vector gradients obtained by differentiation with respect to thickness direction of the element in Ref. [12], Dufva and Shabana [14] applied plane stress conditions to propose a low-order thin plate element with 36DOFs. This gradient efficient element is proved to be more effective and accurate than the higher-order one when the plate is very thin. Matikainen et al. [15] used a set of static tests of small deformation and eigenfrequency analyses to compare the performance of two moderately thick plate elements of the ANCF. Abbas et al. [16] proposed a plate/shell element of variable thickness of the ANCF by modifying the shape function of the element. They also validated the element with several static tests and eigenfrequency analyses, and compared the results with analytic results, experimental data and commercial finite element software. These studies are mainly focused on the establishments of the new plate/shell elements, the descriptions of the elastic forces of the plate element, the applications of the ANCF plate/shell element to solve engineering problems, and et al. However, only a few studies are focus on the dynamic analysis of rotating structures with the ANCF. Zhao et al. [17] used an ANCF thin plate element to analyze the modal characteristics and the eigenvalue loci veering and crossing phenomena. They also derived the analytical formulations of elastic forces and their Jacobians for a thin plate element of ANCF.

This study is organized as follows. In Sec. 2, the ANCF plate/shell element with 48DOFs is briefly introduced. In Sec. 3, the strain energy of the plate element is given and according to the continuum mechanics, the formulations of elastic forces and the stiffness matrix are derived. In Sec. 4, the theory of natural frequency analysis of the rotating plate is described and the analytical expressions of the Jacobian formulation of elastic forces are derived. Numerical examples are presented in Sec. 5. The static deformation of a cantilever plate and the dynamic analysis of a pendulum plate are used to validate the formulations for the plate element of ANCF. The modal characteristics of a plate rotating around the axis across its center are also investigated in this section. The effect of the angular velocities on the modal characteristics of the plate is studied. In Sec. 6, some conclusions are given.

2 ELEMENT KINEMATIC

In previous investigations, a higher-order plate element with 48 degrees of freedom was proposed. Fig.1 shows the global and local coordinates used to define the absolute nodal coordinates used to define the absolute position and gradient coordinates in the absolute nodal coordinate formulation. The position vector \( \mathbf{r} \) of an arbitrary point \( P \) on a thin plate element can be defined using the element shape functions and the nodal coordinate vector as follows

\[
\mathbf{r} = \mathbf{S}(x, y, z) \mathbf{e}
\]

(1)

where \( \mathbf{S} \) is the element shape function matrix expressed in terms of the element spatial coordinates \( x \), \( y \) and \( z \) and \( \mathbf{e} \) is the vector of nodal coordinate. For a plate element with four nodes, each node has 12 nodal coordinates that consist of nodal positions and slope coordinates as follows

\[
\mathbf{e}_p = \begin{bmatrix}
\mathbf{r}_p^T \\
\frac{\partial \mathbf{r}_p^T}{\partial x} \\
\frac{\partial \mathbf{r}_p^T}{\partial y} \\
\frac{\partial \mathbf{r}_p^T}{\partial z}
\end{bmatrix}
\]

(2)
This is a fully parameterized four-node, rectangular, shear deformable plate element with 48 DOFs. Using Fig. 1, the total vector of the element nodal coordinates can be written as follows
\[
e = \begin{bmatrix} e_A^T & e_B^T & e_C^T & e_D^T \end{bmatrix}^T
\] (3)

The shape function is given as
\[
S = \begin{bmatrix} S_A & S_B & S_C & S_D & S_1 & S_2 & S_3 & S_4 & S_5 & S_6 & S_7 & S_8 & S_9 & S_{10} & S_{11} & S_{12} & S_{13} & S_{14} & S_{15} & S_{16} \end{bmatrix}
\] (4)

Where \( I \) is a \( 3 \times 3 \) identity matrix and shape functions are defined as follows
\[
\begin{align*}
S_1 &= (2\bar{\xi}+1)(\bar{\eta}-1)^2 (2\eta+1)(\eta-1)^2, \\
S_2 &= a\bar{\xi}(\bar{\xi}-1)^2 (2\eta+1)(\eta-1)^2, \\
S_3 &= b\eta(\bar{\xi}-1)^2 (2\xi+1)(\eta-1)^2, \\
S_4 &= t\zeta(\bar{\xi}-1)(\eta-1), \\
S_5 &= -\bar{\xi}^2 (2\xi-3)(2\eta+1)(\eta-1)^2, \\
S_6 &= a\bar{\xi}^2 (\bar{\xi}-1)(2\eta+1)(\eta-1)^2 \\
S_7 &= -b\eta\bar{\xi}^2 (2\eta-3), \\
S_8 &= t\zeta\bar{\xi}\eta, \\
S_9 &= -\eta^2\bar{\eta}^2 (2\xi-3)(\eta-1)^2, \\
S_{10} &= -a\eta^2\bar{\eta}^2 (\bar{\xi}-1)(2\eta-3) \\
S_{11} &= -b\eta^2\bar{\eta}^2 (\bar{\xi}-1)^2 (2\eta-3), \\
S_{12} &= t\zeta\bar{\eta}^2 \\
S_{13} &= -\eta^2 (2\bar{\xi}+1)(\bar{\eta}-1)^2 (2\eta-3), \\
S_{14} &= -a\eta^2 (\bar{\xi}-1)^2 (2\eta-3) \\
S_{15} &= b\eta^2 (\bar{\xi}-1)^2 (2\bar{\eta}+1)(\eta-1), \\
S_{16} &= -t\eta\zeta (\bar{\xi}-1)
\end{align*}
\] (5)

where \( \bar{\xi} = x/a, \eta = y/b, \zeta = z/t, a, b \) and \( t \) are the length, width and thickness of the plate in the initial undeformed configuration, respectively.

![Figure 1. The four-node ANCF plate element.](image)

### 3 FORMULATION OF THE ELASTIC FORCES

#### 3.1 Strain energy

For the plate element based on the ANCF, the vector of the element elastic forces can be derived using the strain energy function.
\[
U^e = \frac{1}{2} \int_{V_0} \varepsilon^T \mathbf{E} \varepsilon dV_0
\] (6)

where \( V_0 \) is the element volume in the initial configuration. \( \varepsilon \) is the strain vector the plate. \( \mathbf{E} \) is the elastic coefficient matrix, and for the homogeneous and isotropic material, it is defined as
\[
\mathbf{E} = \begin{bmatrix}
\lambda + 2\mu & \lambda & \lambda & 0 & 0 & 0 \\
\lambda & \lambda + 2\mu & \lambda & 0 & 0 & 0 \\
\lambda & \lambda & \lambda + 2\mu & 0 & 0 & 0 \\
0 & 0 & 0 & \mu & 0 & 0 \\
0 & 0 & 0 & 0 & \mu & 0 \\
0 & 0 & 0 & 0 & 0 & \mu
\end{bmatrix}
\] (7)

where \( \lambda = E\nu/(1+\nu)(1-2\nu) \) and \( \mu = E/2(1+\nu) \) are the Lame’s constants of material. Here, \( E \) is the Young modulus of the material, \( \nu \) is the Poisson’s ratio.
Nonlinear Green-Lagrange strain measure is employed in order to account for geometrically nonlinear behavior and to ensure zero strain under rigid body motion. Strain components are expressed using gradients obtained by differentiation with respect to global spatial coordinates. For plane stress conditions, the deformation gradient is defined by

\[
\begin{bmatrix}
S_{1x}e & S_{1y}e & S_{1z}e \\
S_{2x}e & S_{2y}e & S_{2z}e \\
S_{3x}e & S_{3y}e & S_{3z}e 
\end{bmatrix}
\]

Here, \( S_{ij} = \partial S_j / \partial x_i \), where \( S_{ij} \) is the \( i \)-th row variables in the shape function of the element. \( r \) and \( X \) are the current and initial position of the node as follows

\[
\begin{bmatrix}
[1] \\
[2] \\
[3] 
\end{bmatrix}
= \begin{bmatrix}
r_1 \\
r_2 \\
r_3 
\end{bmatrix}, \quad \begin{bmatrix}
X \\
Y \\
Z 
\end{bmatrix}
= Se
\]

where \( Se \) is the vector of nodal coordinates in the initial undeformed configuration, \( J_0 \) is a constant transformation matrix and is the identity matrix if the element coordinate system is assumed to be initially parallel to the global coordinate system. Therefore the Lagrangian strain tensor can be defined using the matrix of deformation gradient \( J \) as follows

\[
\varepsilon = \frac{1}{2} ( J^T J - I)
\]

where \( I \) is the \( 3 \times 3 \) identity matrix, \( \varepsilon_1 = \frac{1}{2} ( J^T J - I) \) is a symmetric matrix which can be written in a vector form as follows

\[
\varepsilon = \begin{bmatrix}
\varepsilon_1 \\
\varepsilon_2 \\
\varepsilon_3 \\
\varepsilon_4 \\
\varepsilon_5 \\
\varepsilon_6 
\end{bmatrix}
\]

where \( \varepsilon_1 = (e^T S_a e - 1)/2 \), \( \varepsilon_2 = (e^T S_b e - 1)/2 \), \( \varepsilon_3 = (e^T S_c e - 1)/2 \), \( \varepsilon_4 = (e^T S_d e)/2 \), \( \varepsilon_5 = (e^T S_e e)/2 \), \( \varepsilon_6 = (e^T S_f e)/2 \). In the equation above, \( \varepsilon_1 \), \( \varepsilon_2 \) and \( \varepsilon_3 \) are the normal strain components in \( x \), \( y \) and \( z \) directions, respectively, and \( \varepsilon_4 \), \( \varepsilon_5 \) and \( \varepsilon_6 \) denote the shear strains.

### 3.2 Formulation of elastic force

The generalized elastic force vector can be expressed by the partial derivative of the strain energy with respect to the nodal coordinates

\[
Q^e = \frac{\partial U^e}{\partial \varepsilon} = \frac{1}{2} \int_B \left( \frac{\partial \varepsilon}{\partial \varepsilon} \right)^T E \varepsilon dV_0
\]

According to Eqn. (12), the formulation of the elastic force \( Q^e \) is

\[
Q^e = \frac{\partial U^e}{\partial \varepsilon} = K^e \varepsilon
\]

where \( K^e \) is the plane stiffness matrix which can be written as

\[
K^e = (\lambda + 2\mu) K^e_1 + \lambda K^e_2 + 4\mu K^e_3
\]

where

\[
K^e_1 = \frac{1}{4} \int_B \left[ S_{ai} \left( e^T S_a e - 1 \right) + S_{ai} \left( e^T S_b e - 1 \right) + S_{ai} \left( e^T S_c e - 1 \right) \right] dV_0,
\]

\[
K^e_2 = \frac{1}{4} \int_B \left[ (S_{ai} + S_{ai}) e^T S_a e - 1 + (S_{ai} + S_{ai}) e^T S_b e - 1 + (S_{ai} + S_{ai}) e^T S_c e - 1 \right] dV_0,
\]

\[
K^e_3 = \frac{1}{4} \int_B \left[ S_{ai} (e^T S_a e) + S_{ai} (e^T S_b e) + S_{ai} (e^T S_c e) \right] dV_0.
\]
$$\mathbf{S}_{\alpha\beta} = \mathbf{S}_{\alpha} + \mathbf{S}_{\beta}^T, \quad \mathbf{S}_{\alpha} = \mathbf{S}_{\alpha} + \mathbf{S}_{\beta}^T, \quad \mathbf{S}_{\beta} = \mathbf{S}_{\alpha} + \mathbf{S}_{\beta}.$$ (16)

4 NATURAL FREQUENCY OF A ROTATING PLATE

This section presents the modal analysis of a rotating thin-walled structure. The rotating thin-walled structure is shown in Fig. 2. The plate rotates with the speed $\Omega$ around the axis across its center in the $z$ direction, as shown in Fig. 2.

![Figure 2. The rotating thin-walled structure](image)

To analyze the natural frequencies of the system conveniently, the equivalent transformation of the model is introduced. The rotating thin-wall structure is equivalently treated as a nonrotating thin plate subject to an external force, the magnitude of which is equal to the equivalent centrifugal force at a specific rotating speed. The formulation for evaluating the artificially introduced external forces $\mathbf{Q}_e$ can be obtained as follows with the help of the principle of virtual work:

$$\int_{V_0} \rho \dddot{x} \Omega^2 \rho \dddot{x} \Omega^2 0_{d_2} \rho \dddot{r} dv_0 = \mathbf{Q}_e^T \delta \mathbf{e}$$ (17)

here, $\Omega$ is the rotating angular velocity of the plate. In this case, the origin of the global coordinate system is assumed to locate at the rotating center of the plate, therefore the external force is along the plate axial direction. $\dddot{x}$ denotes the distance between the integral point and the rotating axis which can be expressed as

$$\dddot{x}_1 = r_1 = \mathbf{S}_i \mathbf{e}, \quad \dddot{x}_2 = r_2 = \mathbf{S}_i \mathbf{e}.$$ (18)

here, $r_1$ and $r_2$ are the first two components of the global position vector $\mathbf{r}$ of the point, $\mathbf{S}_i$ is the $i^{th}$ row of the shape function. The introduced external force vector can be then expressed as

$$\mathbf{Q}_e = \rho \dddot{x} \Omega^2 \int_{V_0} \mathbf{S}^T \left[ \mathbf{S}_1 \mathbf{e} \quad \mathbf{S}_2 \mathbf{e} \quad 0 \right]^T \rho \dddot{r} dv_0$$ (19)

In order to analyze the natural frequency of the equivalent system, a linearization is performed for the model obtained by using the ANCF. The perturbation form of the dynamic equations of the system is obtained as

$$\mathbf{M} \dddot{\mathbf{e}} + \mathbf{K}_f \delta \mathbf{e} = 0$$ (20)

Where $\mathbf{M}$ is the constant mass matrix of the system which can be obtained by calculating the kinetic energy $T$ of the system as

$$T = \frac{1}{2} \int_{V_0} \rho \dddot{x} \Omega^2 \rho \dddot{x} \Omega^2 0_{d_2} \rho \dddot{r} dv_0 = \frac{1}{2} \int_{V_0} \rho (\dddot{\mathbf{e}} \mathbf{e})^T (\dddot{\mathbf{e}} \mathbf{e}) dv_0 = \frac{1}{2} \dddot{\mathbf{e}}^T \mathbf{M} \dddot{\mathbf{e}}$$ (21)

The mass matrix $\mathbf{M}$ is constant and as a consequence, the centrifugal and Coriolis inertia force vectors are identically zero. $\mathbf{K}_f$ is the system tangential stiffness matrix which is a function of $\mathbf{e}$ defined at the static equilibrium configuration determined by solving the static equilibrium equation

$$\mathbf{Q}_e - \mathbf{0} = 0$$ (22)

Thus, the tangential stiffness matrix of the system can be written as

$$\mathbf{K}_f = \frac{\delta \mathbf{Q}_e}{\delta \mathbf{e}} - \frac{\delta \mathbf{Q}_e}{\delta \mathbf{e}}$$ (23)

Here, the perturbation of the generalized elastic force $\mathbf{Q}_e$ is...
\[
\frac{\partial \mathbf{Q}_e}{\partial \mathbf{e}} = \frac{\partial \mathbf{Q}_e}{\partial \mathbf{e}} = \frac{\partial \mathbf{Q}}{\partial \mathbf{e}}
\]  

(24)

Here, \(\partial \mathbf{Q}_e/\partial \mathbf{e}\) is the Jacobian formulation of the generalized elastic force. Substituting the generalized elastic force \(\mathbf{Q}^e\) obtained from Eqn. (13) into Eqn. (24), the components of the Jacobian formulation for the elastic force \(\mathbf{Q}^e\) can be derived

\[
\frac{\partial \mathbf{Q}^e}{\partial \mathbf{e}} = (\lambda + 2\mu)(\mathbf{K}^e_{11} + \mathbf{K}^e_{12}) + \lambda(\mathbf{K}^e_{m1} + \mathbf{K}^e_{n2}) + 4\mu(\mathbf{K}^e_{m1} + \mathbf{K}^e_{n2})
\]

(25)

where

\[
\begin{align*}
\mathbf{K}^e_{11} &= \frac{1}{4} \int_{V_0} \left[ \mathbf{S}_{a1} \mathbf{S}_d^T \mathbf{S}_d^{11} + \mathbf{S}_{b1} \mathbf{S}_d^T \mathbf{S}_d^{11} + \mathbf{S}_{c1} \mathbf{S}_d^T \mathbf{S}_d^{11} \right] dV_0, \\
\mathbf{K}^e_{12} &= \frac{1}{4} \int_{V_0} \left[ \mathbf{S}_{a1} (\mathbf{e}^T \mathbf{S}_d e - 1) + \mathbf{S}_{b1} (\mathbf{e}^T \mathbf{S}_d e - 1) + \mathbf{S}_{c1} (\mathbf{e}^T \mathbf{S}_d e - 1) \right] dV_0, \\
\mathbf{K}^e_{m1} &= \frac{1}{4} \int_{V_0} \left[ \mathbf{S}_{a1} \mathbf{e}^T \mathbf{S}_d^{11} + \mathbf{S}_{b1} \mathbf{e}^T \mathbf{S}_d^{11} + \mathbf{S}_{c1} \mathbf{e}^T \mathbf{S}_d^{11} + \mathbf{S}_{d1} \mathbf{e}^T \mathbf{S}_d^{11} + \mathbf{S}_{b1} \mathbf{e}^T \mathbf{S}_d^{11} \right] dV_0, \\
\mathbf{K}^e_{n2} &= \frac{1}{4} \int_{V_0} \left[ (\mathbf{S}_{a1} + \mathbf{S}_{a1})(\mathbf{e}^T \mathbf{S}_d e - 1) + (\mathbf{S}_{b1} + \mathbf{S}_{b1})(\mathbf{e}^T \mathbf{S}_d e - 1) \right] dV_0, \\
\mathbf{K}^e_{m1} &= \frac{1}{4} \int_{V_0} \left[ \mathbf{S}_{d1} \mathbf{e}^T \mathbf{S}_d^{11} + \mathbf{S}_{d1} \mathbf{e}^T \mathbf{S}_d^{11} + \mathbf{S}_{d1} \mathbf{e}^T \mathbf{S}_d^{11} \right] dV_0, \\
\mathbf{K}^e_{n2} &= \frac{1}{4} \int_{V_0} \left[ \mathbf{S}_{d1} (\mathbf{e}^T \mathbf{S}_d e) + \mathbf{S}_{d1} (\mathbf{e}^T \mathbf{S}_d e) + \mathbf{S}_{d1} (\mathbf{e}^T \mathbf{S}_d e) \right] dV_0.
\end{align*}
\]

(26)

The symbols in the equations above are defined as the same as in previous sections. The general solution of the system of Eqn. (20) can be assumed as

\[
\delta \mathbf{e} = \mathbf{A} e^{j\omega t}
\]

(27)

Use of this solution leads to the generalized eigenvalue problem

\[
(\mathbf{K}_T - \omega^2 \mathbf{M}) \mathbf{A}_n = 0
\]

(28)

The natural frequencies and associated mode shapes can be determined by solving the preceding equation.

5 Numerical Examples

Four tests have been performed to investigate the performance of the ANCF plate element. All the formulations and the computations are implemented using Matlab. The results obtained are compared with the analytic results, experimental data and ABAQUS commercial finite-element software.

5.1 Static Deformation of a Cantilever Thin Plate.

To validate the derived formulations for the plate element of the ANCF, a static deformation analysis of a cantilever thin plate subjected to one force at the free end is studied. One side of the plate structure is fixed to the ground while the structure is subjected to a force of 50N and 50000N at point A, respectively. The length, width and thickness of the plate are 1.0, 1.0, and 0.01 m, respectively. The material of the plate structure is assumed to be isotropic, Young’s modulus of the material is \(10^{22.07} \text{ N/m}^2\) and Poisson’s ratio 0. The plate is divide into \(1^2, 2^2, 4^2,\) and \(8^2\) elements, respectively. The displacement of point A obtained by using the ANCF element and commercial finite element software ABAQUS are shown in Table 1.

| Table 1. Deformations of the end point |
|------------------|------------------|------------------|------------------|
| Force            | ANCF             | ABAQUS           | ANCF             | ABAQUS           |
| Element          | 50N              | 50000N           | 50N              | 50000N           |
| 1x1              | -0.69382         | -1.0623          | -0.0616          | -0.6471          |
| 2x2              | -0.91783         | -1.2475          | -0.1718          | -0.6559          |
| 4x4              | -1.1359          | -1.2932          | -0.4225          | -0.6585          |
| 8x8              | -1.2817          | -1.3011          | -0.6113          | -0.6682          |
As shown in Table 1, with the increase of the element number, the displacement of the ANCF model is quite close to the result of ABAQUS. The numerical results obtained in this paper are also in a reasonable agreement with those obtained by Mikkola and Matikainen [18].

5.2 Simple Plate Pendulum Example

In this example, a simple pendulum structure is used to demonstrate that the ANCF element fulfills the requirement of energy conservation. The length, width and thickness of the plate pendulum are 1.0, 1.0, and 0.1m, respectively. Here, the modulus of elasticity is reduced to $1.0 \times 10^{10}$ N/m², while Poisson’s ratio is assumed to be zero. The plate is connected to the ground using a spherical joint and is discretized by using one element only. The equations of motion as

$$\mathbf{M}\ddot{\mathbf{e}} + \mathbf{K}\mathbf{e} = \mathbf{F}_{\text{ext}}$$  \hspace{1cm} (29)

where $\mathbf{F}_{\text{ext}}$ is the vector of generalized external nodal force. Using the equation of motion above and numerical integration procedure, the pendulum responses can be predicted. Fig. 3 shows the simulation of the plate motion under the effect of gravity. Since the modulus of elasticity is very small, the deformation of the pendulum is obvious.

5.3 Modal Analysis of a completely free Thin-walled Structure

The modal analysis of a square thin plate with completely free boundaries is studied in this section. Although somewhat idealized, this case has a number of advantages: there are no kinematic boundary conditions and it demonstrates the possibility of the element to describe the rigid bodies’ motions. The square plate has geometrical dimensions of $a = b = 1$m and $h = 0.01$m. The material is linearly isotropic elastic with a modulus of elasticity $E = 7.3 \times 10^{10}$ N/m², Poisson’s ratio $\nu = 0.3$, and volumetric mass density $\rho = 2780$kg/m³. The first sixteen dimensionless natural frequencies $f = \omega/\omega_b$ of the plate modeled by different numbers of plate elements of ANCF compared with the results of free boundary are shown in Table 1. The numbers in the first row of the table denote the element number used to model the plate. The natural frequencies are normalized by using the frequency $\omega_b = \pi^2 \sqrt{D/(\rho h^3)}$, where $D = Eh^3/12(1-\nu^2)$. It can be found from Table 1 that the numerical results converge rapidly to the analytical results with an increase of the number of plate elements. It can also be found from Table 2 that because of free boundary conditions, the first six natural frequencies of the unconstrained plate from mode 1 to 6 are very close to zero.

5.4 Modal Analysis of a constrained Thin-Walled Structure

A plate which could rotate around the axis across its center in the $z$ direction, as shown in Fig. 2, is used in this test. The geometry parameters and material properties of the plate are the same as those in the previous sections. All degrees of freedom except the one that rotating the $z$-axis of the center point of the plate are fixed. Table 3 shows the first ten resonant frequencies in Hertz obtained from ANCF.
element and the commercial software ABAQUS. In the ABAQUS software, the results are obtained by employing the S4R element. As shown in Table 3, since only 5 degrees of freedom of the center of the plate is constrained, the first natural frequency is equal to zero. With the increase of the element number, the numerical results converge slowly to the software’s results, while all the numerical results are higher than the software’s results.

| Table 2. First 16 natural frequencies of the free square plate modeled by the ANCF |
|-----------------|-----------|-----------|-----------|-----------|-----------|-----------|
| Modal | 1x1 | 2x2 | 4x4 | 8x8 | 16x16 | Analytical |
| 1 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 2 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 3 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 4 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 5 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 6 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 7 | 1.4383 | 1.4380 | 1.4244 | 1.3897 | 1.3785 | 1.3646 |
| 8 | 2.2738 | 2.2737 | 2.0761 | 2.0635 | 2.0346 | 1.9855 |
| 9 | 3.5947 | 3.9945 | 2.9253 | 2.8861 | 2.7987 | 2.4591 |
| 10 | 71.9937 | 7.9945 | 7.8878 | 7.4175 | 6.7932 | 6.1900 |
| 11 | 71.9937 | 7.9945 | 7.8878 | 7.4175 | 6.8774 | 6.1900 |
| 12 | 83.5548 | 18.0356 | 14.9193 | 7.4175 | 6.1900 | 6.1900 |
| 15 | 93.22964 | 72.2273 | 16.6869 | 7.4175 | 6.8774 | 6.1900 |
| 16 | 124.2336 | 72.5239 | 17.6653 | 9.7514 | 8.8631 | 7.8191 |

| Table 3. First 10 natural frequencies (in Hertz) of the constrained square plate |
|-----------------|-----------|-----------|-----------|-----------|-----------|
| Modal | ANCF Element | ABAQUS |
| 1 | 0.0000 | 3.4312 | 3.6152 | 4.1791 | 4.3791 | 4.4398 | 5.1103 |
| 2 | 3.4424 | 4.9895 | 3.9866 | 3.7673 | 3.6757 | 1.5316 |
| 3 | 4.9430 | 5.1908 | 4.8177 | 4.5738 | 4.3564 | 2.8150 |
| 4 | 5.7255 | 5.4087 | 4.8118 | 4.5739 | 4.3565 | 3.2832 |
| 5 | 5.7255 | 6.3029 | 5.1077 | 5.0366 | 4.9807 | 4.8278 |
| 6 | 7.1908 | 7.0456 | 5.4927 | 5.1908 | 4.9807 | 4.8278 |
| 7 | 383.4185 | 32.8456 | 46.4085 | 31.3646 | 22.8608 | 8.7979 |
| 8 | 383.4185 | 32.8456 | 46.4085 | 31.3646 | 22.8608 | 8.7979 |
| 9 | 984.5857 | 11.608 | 48.5927 | 32.8456 | 22.8608 | 8.7979 |
| 10 | 1234.1353 | 16.963 | 52.1419 | 41.5233 | 32.8456 | 8.7979 |

| 5.5 Modal Analysis of a Rotating Thin-walled Structure |

In this test, the modal characteristics of the rotating plate with different rotation speeds are investigated. The rotation speeds are given as 0 rad/s, 10 rad/s, 20 rad/s, 30 rad/s, 40 rad/s and 50 rad/s, respectively. The first five resonant frequencies of the rotating plate are given in Table 4. The variation of the frequencies of the plate with respect to the angular velocity is given in Fig. 4.

| Table 4. First five resonant frequencies (in Hertz) of the rotating square plate modeled by the ANCF for different angular velocities |
|-----------------|-----------|-----------|-----------|-----------|-----------|
| Modal | ANCF Element | ABAQUS |
| 1 | 3.4312 | 3.6152 | 4.1791 | 4.3791 | 4.4398 | 5.1103 |
| 2 | 4.9848 | 5.1908 | 5.4087 | 5.5947 | 5.6287 | 6.0184 |
| 5 | 5.6142 | 5.8001 | 6.6739 | 6.9909 | 7.0456 | 8.0961 |
As shown in Table 4 and Fig. 4, the frequencies of the plate increase with the increase of the rotation speeds. The reason is the centrifugal force changes the shape of the plate and influences its resonant frequencies. The nodal lines of the lowest five mode shapes of the rotating plate with different angular velocities are given in Fig. 5. The phenomenon which is called the eigenvalue loci veering phenomenon in engineering systems can be observed in Fig. 5. When the angular velocities are between 0 rad/s and 40 rad/s, the orientations of the first and the second eigenvalue loci change with the variation of the angular velocities, the third to the fifth eigenvalue loci do not change very much. However, when the angular velocity increases to 50 rad/s, the mode switching happens and the eigenvalue trajectories very much. The angular velocity and the centrifugal force have a significant influence on the dynamic behavior of the rotating plate.

Figure 4. Lowest five natural frequencies versus angular velocity

As shown in Table 4 and Fig. 4, the frequencies of the plate increase with the increase of the rotation speeds. The reason is the centrifugal force changes the shape of the plate and influences its resonant frequencies. The nodal lines of the lowest five mode shapes of the rotating plate with different angular velocities are given in Fig. 5. The phenomenon which is called the eigenvalue loci veering phenomenon in engineering systems can be observed in Fig. 5. When the angular velocities are between 0 rad/s and 40 rad/s, the orientations of the first and the second eigenvalue loci change with the variation of the angular velocities, the third to the fifth eigenvalue loci do not change very much. However, when the angular velocity increases to 50 rad/s, the mode switching happens and the eigenvalue trajectories very much. The angular velocity and the centrifugal force have a significant influence on the dynamic behavior of the rotating plate.

![Diagram of mode shapes with different angular velocities](image-url)
6 CONCLUSIONS

In this study, the dynamic model of the rotating plate is established by using the ANCF. The analytical formulations of elastic forces and their Jacobians for the plate of ANCF are derived. Two cases are employed to validate the derived formulations which are the static deformation test of a cantilever plate and the dynamic test of a pendulum plate. The numerical results shows a good agreement with those obtain in previous publications. The modal analysis of the free, constrained and rotating square plate are studied based on the plate element of ANCF, and the numerical results are compared with that of the analytic method and the finite-element software ABAQUS. The rotating square plate is equivalently treated as a nonrotating one subject to an equivalent external force. The effect of the angular velocity on the natural frequencies of the rotating plate is analyzed. The result shows that the natural frequencies increase with the increase of the angular velocities. The eigenvalue loci veering phenomenon is observed. The angular velocity has a significant influence on the dynamic behavior the rotating plate.

ACKNOWLEDGEMENTS

The authors appreciate the financial support by the National Natural Science Foundation of China (51275292) and State Key Development Program of Basic Research of China (2014CB046600).

REFERENCES


Dynamic Fracture Simulation of Flexible Multibody Systems via Absolute Nodal Coordinate Formulation and SPH Method

Wei Hu, Qiang Tian, Haiyan Hu

School of Aerospace Engineering
Beijing Institute of Technology
5 South Zhongguancun Street, Haidian District, Beijing, 100081, China
huwei732@163.com, tianqiang_hust@aliyun.com, haiyan_hu@bit.edu.cn

ABSTRACT
A new computation methodology is proposed to study the dynamic fracture process of flexible multibody systems with initial cracks. The potential fracture domains of flexible parts are described by using the Smoothed Particle Hydrodynamics (SPH) method, and the other domains are represented by using the Absolute Nodal Coordinate Formulation (ANCF). The interaction forces between the SPH particles and the ANCF elements are transmitted by the virtual particles. The GPU based OpenACC parallel technique is used to improve the efficiency of interaction detection for SPH particles. A predictor-corrector scheme is used to solve the SPH particles governing equations, while the generalized-alpha method is used to solve the huge set of Differential Algebraic Equations (DAEs) of the ANCF elements. The OpenMP parallel technology is also employed to efficiently evaluate the element elastic forces and their Jacobians. Finally, a numerical example is given to validate the proposed computation methodology.

Keywords: Flexible Multibody System, Smoothed Particle Hydrodynamics, Absolute Nodal Coordinate Formulation, Dynamic fracture, Crack.

1 INTRODUCTION
Dynamic fracture phenomena of multibody systems can be often observed in many engineering fields, such as the sudden crack of the belt of pulley-belt driving systems, the instantaneous fracture of long motor shaft. In most cases the dynamic fracture of the multibody parts will directly lead to the collapse of the whole system’s working performance. Thus, to deeply understand and simulate the dynamic fracture process can help to the development of modern mechanical systems.

Over the past years many researchers’ attention has been attracted to the subject on the crack growth due to its importance and challenge [1-4]. Previous studies indicate [5] that the nature of the dynamic fracture phenomena involves a complex crack growth process. However, most previous studies were mainly focused on the crack growth process of a single structure part [3, 4]. The relative motion of different parts in the crack growth process was not considered. Thus, the static or quasi-static state is generally assumed in the analysis process. The Finite Element Method (FEM) is one of the most widely used methods to study the crack growth process.

Sukumar and Belytschko [6] proposed an extended Finite Element Method (XFEM). Using the XFEM, the cracks are completely independent of the mesh, and then there is no need to assure the consistency between the finite element edges and the cracks. Thus, the XFEM can simulate the growth process of arbitrary branched cracks. However, the XFEM cannot be directly used for the dynamic fracture analysis of flexible multibody systems with initial cracks.

Another alternative method, the mesh-free Smoothed Particle Hydrodynamics (SPH) method [7], has also been successfully used to simulate fracture problems [8, 9]. Das and Cleary [10] studied the influence of rock shapes on brittle fracture behaviors by using the SPH method. The obtained numerical results are in a good agreement with experiment results. Chakraborty and Shaw [11] developed a new strategy to model the structure crack initiation and crack growth
using the SPH method. The crack path was successfully captured by the SPH method. However, there are still some tough problems that restrict the development of SPH, including the numerical accuracy, tensile instability and the disordered particle distribution [12]. To avoid or attenuate these problems, many SPH correction schemes are proposed, such as the Reproducing Kernel Particle Method (PKPM) [13] for improving the particle approximation and the Corrective Smoothed Particle Method (CSPM) [14] for improving the result accuracy for the particles near boundaries. Also, the artificial stress method proposed by Monaghan and Gray [15, 16] can be used to attenuate the tensile instability problems and preserve the numerical stabilization. However, the particle disorder distribution problem is still an open and challenging problem in the field of SPH [12].

By considering the advantages of both traditional FEM and SPH, recently the FEM-SPH coupling method has been widely used for the fracture problem analysis [17, 18]. Chuzel-Marmot et al. [18] studied the fracture process of concrete slabs under high-speed impacts. The SPH particles are used to discretize the potential fracture domain while the finite elements are used to mesh the other domains. To connect the finite elements and the SPH particles, a three-dimensional FEM-SPH gluing way is proposed. From previous studies, the FEM-SPH coupling method can be considered as a promising method for the dynamic fracture simulation of flexible multibody systems with initial cracks. However, using the traditional incremental FEM, the dynamics of flexible multibody systems undergone large deformation and rotation cannot be accurately captured. The Absolute Nodal Coordinate Formulation (ANCF), which was initially proposed by Shabana [19], has been regarded as a benchmark in the development of flexible multibody dynamics [20, 21]. The interested readers on the unique features and recent developments of ANCF can refer to the review work by Gerstmayr et al. [22]. The ANCF-SPH coupling method has already been used to study the coupling dynamics of liquid filled flexible multibody system by the authors [23].

In this study, based on the authors’ previous work [23], a new computation methodology to study the dynamic fracture of a flexible multibody system with initial crack is proposed. The SPH method is used to model the potential fracture domains of flexible components, while the finite elements of ANCF are used to mesh the other domains. The remaining parts of the paper are organized as follows. In Section 2, the approximation of a continuous solid by using SPH particles is briefly introduced. A fracture model and a simple contact model for SPH particles are described in Section 3. In Section 4, the used finite elements of ANCF are revisited. In Section 5, the coupling technique for connecting SPH particles and ANCF elements are described. Section 6 gives the computation strategies including the generalized-alpha method and a two-step predictor-corrector. The whole computation process is parallelized by using OpenMP and OpenACC derivatives. Finally, to validate the proposed computation methodology a numerical example is given in Section 7.

2 SPH FORMULATIONS FOR THE SOLID PROBLEM

According to the SPH method [12], a continuous function vector \( \mathbf{f}(\mathbf{r}) \) can be approximated by

\[
\mathbf{f}(\mathbf{r}) \approx \int_{\Omega} \mathbf{f}(\mathbf{x}) W(\mathbf{r} - \mathbf{x}, h) d\mathbf{x}, \tag{1}
\]

where \( \mathbf{r} \) is a vector in three-dimensional space, \( \Omega \) is the integration domain, the scalar function \( W \) is the smoothing function, \( h \) denotes the smoothing length of the smoothing function. In the SPH method, the integration domain \( \Omega \) is meshed via a number of non-connected particles, as shown in Figure 1. The spherical domain centered at a particle \( a \) is defined as the support domain of the particle \( a \), and \( k \) is a constant related to the smoothing function used.
Based on the continuum mechanics, the momentum equations [12] for an arbitrary solid particle $a$ can be written as

$$\frac{dv_a^\alpha}{dt} = \frac{1}{\rho_a} \frac{\partial \sigma_a^{\alpha\beta}}{\partial r_a^\beta}, \quad (2)$$

where $\rho_a$ is the density of the particle $a$, $v_a^\alpha$, $r_a^\beta$, and $\sigma_a^{\alpha\beta}$ are the components of the velocity vector $v_a$, the position vector $r_a$, and the Cauchy stress tensor $\sigma_a$ at the particle $a$ respectively, and $\alpha = x, y, z$ and $\beta = x, y, z$. The Cauchy stress tensor can be expressed by

$$\sigma_a^{\alpha\beta} = -p_a \delta^{\alpha\beta} + \tau_a^{\alpha\beta}, \quad (3)$$

where $p_a$ is the isotropic pressure, and $\tau_a^{\alpha\beta}$ are the components of the shear stress $\tau_a$. The isotropic pressure $p_a$ in Equation (3) can be calculated by the state equation written as

$$p_a = c_a^2 (\rho_a - \rho_0), \quad (4)$$

where $\rho_0$ is the reference density, $c_a$ is the speed of sound at the particle $a$. $c_a$ can be calculated by

$$c_a = \sqrt{\frac{K}{\rho_a}}, \quad (5)$$

where $K$ denotes the material bulk modulus. In addition, according to the law of conservation of mass the continuity equation can be written as

$$\frac{d\rho_a}{dt} = -\rho_a \frac{\partial v_a^\alpha}{\partial r_a^\beta}. \quad (6)$$

The linear elastic relation between the stress and strain tensors [15, 16] is used so as to obtain the rate of change for $\tau_a$, given as

$$\frac{d\tau_a^{\alpha\beta}}{dt} = 2G \left( \varepsilon_a^{\alpha\beta} - \frac{1}{3} \delta^{\alpha\beta} \varepsilon_a^{\gamma\gamma} \right) + \tau_a^{\alpha\gamma} \delta_a^{\gamma\beta} + \tau_a^{\beta\gamma} \delta_a^{\gamma\alpha}, \quad (7)$$

where $\varepsilon_a^{\alpha\beta}$ are the components of the strain rate tensor $\varepsilon_a$, which is defined as

$$\varepsilon_a^{\alpha\beta} = \frac{1}{2} \left( \frac{\partial v_a^\alpha}{\partial r_a^\beta} + \frac{\partial v_a^\beta}{\partial r_a^\alpha} \right), \quad (8)$$

and $\delta_a^{\alpha\beta}$ are the components of the rotation rate tensor $\delta_a$, which is defined as
\[
\delta_{\alpha}^{\nu} = \frac{1}{2} \left( \frac{\partial v_{\alpha}^{\nu}}{\partial t} - \frac{\partial v_{\nu}^{\alpha}}{\partial t} \right),
\]

where \( \gamma = x, y, z \). The SPH form of Equations (2), (6), (8) and (9) are

\[
\frac{dv_{\alpha}^{\nu}}{dt} = \sum_{b=1}^{N} m_{b} \left( \frac{\sigma_{\alpha}^{\nu}}{\rho_{a}} + \frac{\sigma_{\nu}^{\alpha}}{\rho_{b}} + \Pi_{ab} \delta_{\alpha}^{\nu} \right) \nabla_{\beta} W_{ab},
\]

\[
\frac{d\rho_{\alpha}}{dt} = \sum_{b=1}^{N} m_{b} \nu_{ab} \nabla_{\alpha} W_{ab},
\]

\[
\dot{\varepsilon}_{\alpha}^{\nu} = \frac{1}{2} \sum_{b=1}^{N} \frac{m_{b}}{\rho_{b}} \left( \nu_{\alpha}^{\nu} \nabla_{\beta} W_{ab} + \nu_{\beta}^{\nu} \nabla_{\alpha} W_{ab} \right),
\]

\[
\dot{\sigma}_{\alpha}^{\nu} = \frac{1}{2} \sum_{b=1}^{N} \frac{m_{b}}{\rho_{b}} \left( \nu_{\beta}^{\nu} \nabla_{\alpha} W_{ab} - \nu_{\alpha}^{\nu} \nabla_{\beta} W_{ab} \right),
\]

where \( \nabla_{\beta} W_{ab} = \frac{r_{ab}^{\alpha}}{r_{ab}^{\nu}} \frac{\partial W_{ab}}{\partial \bar{r}_{ab}} \), and \( \Pi_{ab} \) is the introduced artificial viscosity term [24] between the particle \( a \) and \( b \).

### 3 SPH Fracture Model

In order to describe the particle fracture phenomenon, the pseudo-spring method proposed by Chakraborty and Shaw [11] is used in this study. Only the closely linked neighboring particles in the support domain of the particle \( a \) are chosen for the computation. As shown in Figure 2, the closely linked two neighboring particles are defined as those with distances of \( \Delta D \) (initial particle distance) and \( \sqrt{2} \Delta D \) to each other at the initial time. For three-dimensional problems, particles with a distance of \( \sqrt{3} \Delta D \) should also be considered.

![Figure 2. View of the closely linked particles of the particle \( a \)](image)

According to the pseudo-spring method [11], the crack growth process in the material involves a damage growth process in the pseudo-spring. The damage growth is determined by the relative stress in the pseudo-spring. To measure the stress magnitude in the pseudo-spring of the linked two neighboring particles \( a \) and \( b \), the relative stress \( \bar{\sigma}_{ab} \) in the pseudo-spring is defined as
\[ \bar{\sigma}_{ab} = \frac{\bar{\sigma}_a \rho_b c_b + \bar{\sigma}_b \rho_a c_a + \rho_a \rho_b c_a c_b v_{ab}}{\rho_a c_a + \rho_b c_b} , \]  

(14)

where

\[ \bar{\sigma}_a = n_{ab}^T \sigma_a n_{ab} , \quad \bar{\sigma}_b = n_{ab}^T \sigma_b n_{ab} , \]

(15)

\[ v_{ab} = n_{ab} \cdot v_{ab} = \frac{1}{r_{ab}} \left( r_{ab}^2 v_{ab}^2 + r_{ab}^2 v_{ab}^2 + r_{ab}^2 v_{ab}^2 \right) , \]

(16)

\[ n_{ab} = \frac{r_{ab}}{r_{ab}} \left[ r_{ab}^T \quad r_{ab}^T \right]^T . \]

(17)

According to the Rankine criterion, the damage in the pseudo-spring is considered to be initiated if the distance between the two particles meet the condition,

\[ \frac{r_{ab} - \Delta D}{\Delta D} \geq \frac{\sigma_{\text{max}}}{E} , \]

(18)

where \( \sigma_{\text{max}} \) is the critical maximum stress, \( E \) is the Young’s modulus, \( r_{ab} \) denotes the distance between the two neighboring particles. Once the damage in the pseudo-spring initiates, the crack opening displacement (\( \Delta L \)) increases linearly. The critical stress \( \sigma_{\text{cri}} \) equals to the maximum stress \( \sigma_{\text{max}} \) when the damage is just initiated, and decreases to zero when the crack opening displacement increases to the maximum value (\( \Delta L_{\text{max}} \)). The crack opening displacement between the two particles can be written as

\[ \Delta L = r_{ab} - r_{ab0} , \]

(19)

where \( r_{ab0} \) is the critical distance according to Equation (18) when the damage is just initiated.

The state of the pseudo-spring damage can be determined by

\[ \begin{cases} 
\text{if : } \bar{\sigma}_{ab} > \sigma_{\text{cri}} & \text{damage growth} \\
\text{if : } \bar{\sigma}_{ab} \leq \sigma_{\text{cri}} & \text{no damage growth} 
\end{cases} \]

(20)

With the growth of the damage in the pseudo-spring, the interaction force between the linked two particles \( a \) and \( b \) would no longer be the same as the undamaged pseudo-spring. To reflect the attenuated interaction forces in the pseudo-springs, a new parameter \( I_{ab} \), named as ‘interaction factor’ [11], is defined as

\[ \begin{align*}
I_{ab} &= \frac{\sigma_{\text{cri}}}{\bar{\sigma}_{ab}} & (\Delta L < \Delta L_{\text{max}}) \\
I_{ab} &= 0 & (\Delta L \geq \Delta L_{\text{max}})
\end{align*} \]

(21)

For the undamaged pseudo-spring, \( I_{ab} = 1 \), while for the fully damaged one, \( I_{ab} = 0 \). Inserting Equation (21) into Equations (10-13) yields

\[ \frac{dv}{dt} = \sum_{b=1}^{N_b} \left( \frac{\sigma_{ab}^{\text{eff}}}{\rho_b^2} + \Pi_{ab} \Omega_{ab}^{\text{eff}} \right) v_{ab}^a W_{ab} + \sum_{b=1}^{N_b} \left( \frac{\sigma_{ab}^{\text{eff}}}{\rho_a^2} + \Pi_{ab} \Omega_{ab}^{\text{eff}} \right) v_{ab}^b W_{ab} , \]

(22)

\[ \frac{d\rho}{dt} = \sum_{b=1}^{N_b} m_b \rho_b \nabla_{ab} v_{ab}^a W_{ab} + \sum_{b=1}^{N_b} m_b \rho_b \nabla_{ab} v_{ab}^b W_{ab} , \]

(23)

\[ l_{ab}^{\text{eff}} = \frac{1}{2} \sum_{b=1}^{N_b} m_b \left( \rho_{ab} \nabla_{ab}^a W_{ab} + \nabla_{ab} v_{ab}^a W_{ab} \right) + \frac{1}{2} \sum_{b=1}^{N_b} m_b \left( \rho_{ab} \nabla_{ab}^b W_{ab} + \nabla_{ab} v_{ab}^b W_{ab} \right) , \]

(24)
\[ \dot{\theta}_{ab} = \frac{1}{2} \sum_{b=1}^{N_b} \frac{m_b}{\rho_b} \left( \theta_{ab}^\alpha \nabla_{\theta} W_{ab} - \theta_{ab}^\beta \nabla_{\beta} W_{ab} \right) + \frac{1}{2} \sum_{b=1}^{N_b} \frac{I_{ab}}{\rho_b} \left( \theta_{ab}^\alpha \nabla_{\theta} W_{ab} - \theta_{ab}^\beta \nabla_{\beta} W_{ab} \right), \]  

where \( N_1 \) is the number of particles with the ‘interaction factor’ equal to 1 in the support domain of the particle \( a \), while \( N_2 \) is the number of other particles with the ‘interaction factor’ lower than 1.

4 FULLY PARAMETRIC BEAM ELEMENT OF ANCF

The ANCF originally proposed by Shabana [19], is an accurate, non-incremental finite element method to study the dynamics of a flexible multibody system subject to both large overall motion and large deformation [19]. In this study, the fully parametric beam element of ANCF is used to model all the flexible domains far from the cracks. According to the ANCF, the location and deformation of a material point in a finite element are defined in a global coordinate system such that no coordinate transformation is required, the mass matrix remains constant and the centrifugal and Coriolis forces in the finally derived dynamic equations vanish. Such an approach leads to great convenience in computation process. Many finite elements of ANCF have been developed for the dynamic analysis of different flexible parts. In this study, as shown in Figure 3, the classic fully parametric beam element of ANCF is used.

![Figure 3](image.png)

**Figure 3.** The configuration of beam element of ANCF

In the ANCF, the nodal coordinates of a finite element are defined in the global coordinate system, and the global position and velocity vector of an arbitrary point \( P \) in the finite element can be written as

\[ \mathbf{r} = \mathbf{S} \mathbf{e}, \quad \mathbf{v} = \dot{\mathbf{S}} \mathbf{e}, \]

where \( \mathbf{r} \) and \( \mathbf{v} \) are the global position vector and the global velocity vector of an arbitrary point \( P \), respectively. \( \mathbf{S} \) is the shape function matrix of the element [25], \( \mathbf{e} \) and \( \dot{\mathbf{e}} \) are the absolute nodal coordinate vector and velocity vector of the element, respectively.

5 COUPLING TECHNIQUE FOR SPH PARTICLES AND ANCF ELEMENTS

In this study, based on the authors’ previous work [23], a new ANCF-SPH coupled model for flexible multibody systems with initial cracks is proposed. Figure 4 gives the configuration of a beam described by ANCF fully parametric beam elements and SPH particles. As shown in Figure 4, to transmit the interaction forces between the SPH particles and the beam elements, the virtual particles are embedded in the elements close to the connecting interface. The field variables of the virtual particles, such as the particle positions, velocities, densities and stress are consistent with those of the corresponding points of the ANCF elements.

Similarly, the interaction forces applied on the ANCF elements can also be calculated. According to Equation (10), the acceleration of the particle \( a \) can be calculated. Then according to the Newton’s law, the total external forces applied on the particle \( a \) can be further obtained.
Obviously, parts of forces are transmitted from the virtual particles, the red circles marked 1, 2 and 3. Actually, these forces can be regarded as the external forces of the corresponding point of ANCF elements.

Figure 4. The configuration of beam with ANCF elements and SPH particles

6 COMPUTATION STRATEGY

The assembly of the ANCF elements can be easily carried out by transforming the absolute nodal coordinate vector \( \mathbf{e} \) of a finite element into the generalized coordinate vector \( \mathbf{q} \) of the flexible multibody system. Hence, the final dynamic equations for the ANCF based elements of the system can be expressed in a compact form as a set of Differential Algebraic Equations (DAEs) with a constant mass matrix as following [22]

\[
\begin{align*}
\mathbf{M} \dot{\mathbf{q}} + \Phi^\top \lambda + \mathbf{F}(\mathbf{q}) &= \mathbf{Q}(\mathbf{q}, \dot{\mathbf{q}}) \\
\Phi(\mathbf{q}, t) &= 0
\end{align*}
\]

(27)

where \( \mathbf{M} \) is the constant mass matrix of the system, \( \Phi(\mathbf{q}, t) \) represents the vector that contains the system constraint equations, \( \Phi_q \) is the derivative matrix of constraint equations with respect to the generalized coordinate vector \( \mathbf{q} \), \( \lambda \) is the Lagrange multiplier vector. \( \mathbf{F}(\mathbf{q}) \) is the elastic force vector, \( \mathbf{Q}(\mathbf{q}, \dot{\mathbf{q}}) \) is the external force vector. In this study, the generalized-alpha method [26, 27] is used to solve it.

Different from the DAEs for multibody system dynamics, the dynamic equations in the SPH method are a set of ordinary differential equations. In this study, a two-step predictor-corrector scheme [28, 29] which offers the second-order accuracy in time has been adopted in this study. To increase the computation efficiency, a parallel computation methodology based on OpenMP [30] and OpenACC [31] is proposed in this study. The calculation of the elastic force vector and its Jacobi matrix of the ANCF elements during the iteration process can be parallelized by using the OpenMP directives [30]. Considering the large number of cores in the computer Graphics Processing Unit (GPU), both the particle contact detection and the solution of the integral equations can be parallelized by using OpenACC directives. More details about the OpenACC based parallel scheme can be referred to the works on the website [31].

7 NUMERICAL EXAMPLE

As shown in Figure 5, the case is to study the fracture dynamics of a slider-crank mechanism. The slider is a rigid part, and the mass of the slider is set as 50 kg. The connecting-rod (AD) with initial cracks is assumed to be flexible. The initial configuration is chosen as the crank and the connecting rod are collinear. There are three revolute joints at points O, A and D, respectively. The motion of the mechanism is only in \( X-O-Y \) plane, and the rotation axes of the three revolute joints are parallel to axis \( O-Z \). As shown in Figure 5, the whole mechanism will move under the action of a specified moment \( \mathbf{M}_{cr} \). The value of the moment is equal to 4000 Nm when \( t < 1 \) s, and equal to 0 when \( t > 1 \) s.
The connecting rod is divided into three parts, AB, BC and CD. AB and CD are meshed by using ANCF fully parametric beam element described in Section 4, while BC is modelled by using SPH particles. Crank OA is meshed by using ANCF fully parametric beam elements. The virtual particles are used to connect the SPH particles and ANCF full parametric beam elements. The lengths of crank OA, sections AB and CD are 1.0 m, while the length of BC is set as 0.2 m. The cross-section size of the crank and the connecting rod are set as 0.22 m (width) × 0.22 m (height). The material Young’s modulus of the crank and connecting rod are set as 1.0e6 MPa and 1.0e5 MPa, respectively. All material density is set as 8000 kg/m³. The material Poisson’ ratio of the crank and the connecting rod are set as 0.3. The yield stress and the critical maximum stress $\sigma_{\text{max}}$ are set as 2.0e2 MPa and 5.0e2 MPa, respectively. As shown in Figure 5, the lengths of two initial cracks of the connecting rod are taken as 0.05 m, the widths are equal to the initial particle distance $\Delta D$. The directions of the cracks are perpendicular to the axial direction of the connecting rod. The total simulation time is 3.0 s.

![Figure 5. Configuration of the slider-crank mechanism](image)

As shown in Table 1, three ANCF-SPH perfect models (model I, II, III) without cracks on the connecting rod meshed by using different number of particles are comparatively simulated. For result comparison, another ANCF perfect model (model IV) without initial cracks, is also simulated. For this model, the connecting rod and crank are meshed only by using 11 and 5 ANCF fully parametric beam elements, respectively. Finally, to study of the dynamic fracture phenomenon of the system, an ANCF-SPH model with two initial cracks (model V) is also established. All the parameters of model V are same as those of model III.

| Table 1. Parameters and cost CPU time for ANCF-SPH perfect models. |
|-------------------|-------------|-------------|-------------|
| Initial particle distance (m) | Model I | Model II | Model III |
| 0.02 | 0.01 | 0.00667 |
| Smoothing length $h$ (m) | 0.026 | 0.013 | 0.00867 |
| Number of SPH particles | 1331 | 9261 | 29791 |
| Number of virtual particles | 242 | 882 | 1922 |
| Step size (s) | 5.0e-6 | 2.0e-6 | 1.0e-6 |
| Cost CPU time (hours) | 2.75 | 20 | 49 |

Table 1 also gives the cost CPU time for different models. It can be seen from Table 1 that for all the ANCF-SPH perfect models the integration step sizes are very small. Based on the Courant condition, it can be found that the reason that limits the size of integration step is the high velocity of sound calculated according to Equation (5). It can also be found that in previous work by Chakraborty and Shaw [11], the step size was even set as a very small value 5e-7 s for the crack growth simulation of an imperfect slab subject to an impact load. Thus, it can be obviously seen from Table 1 that though the parallel computation technique is adopted, for the ANCF-SPH models the cost CPU time is still very long, especially for the model III. It was also noted that for the ANCF perfect model (model IV) the cost of CPU time is only 2 minutes.
Figure 6 shows the time history of the point D displacement in X-direction for different models. It is clearly shown that for all the perfect models without initial cracks the converged results are achieved by simulating the ANCF-SPH perfect model III. Also, the results curve of the ANCF-SPH perfect model III matches very well with that of ANCF perfect model IV. From Figure 6 it can also be seen that if the simulation time less than 1.2 s the curve of ANCF-SPH model with two initial cracks (model V) is also in a good agreement with those of model III and model IV. However, with simulation time passing by the crack will gradually grow and finally fracture, which lead to significant deviations of the model V results from those of the model III and model IV.

![Figure 6.](image1)

**Figure 6.** The point D displacements in X-direction obtained by using different models.

![Figure 7.](image2)

**Figure 7.** The point D velocities in X-direction obtained by using different models.

Figure 7 shows the time history of the velocity of point D in X-direction for different models, which further indicates that the convergent results can be achieved by simulating the model III. Moreover, it can be clearly seen from Figure 7 that for the model V after the connecting rod fracture ($t > 1.2$ s) the velocity of point D in X-direction will remain constant. After the connecting rod fracture the constant velocity results of point D can be validated by the point D linearly increasing displacements shown in Figure 6.

Figure 8 shows the dynamic configurations of the system with cracks at 4 typical instants. From the scaled view of the cracks it can be found that the cracks will gradually grow from the crack tips under the action of the moment, and after fracture there is a fracture surface with plastic deformation. The total cost CPU time for model V is about 52 hours. Therefore, from all the above analysis it can be concluded that the computation methodology developed in this study is effective for the dynamic fracture simulation of the flexible multibody system with initial cracks.
\[ t = 1.0s \]

\[ t = 1.25s \]

**Figure 8.** Dynamic configurations (Model V)

### 8 CONCLUSIONS

In this study, a new computation methodology is proposed to study the dynamic fracture phenomenon of a flexible multibody system subjected to large overall motions. The Smoothed Particle Hydrodynamics (SPH) method is used to model potential fracture domains and the Absolute Nodal Coordinate Formulation (ANCF) is used to model other domains of a flexible multibody system. The pseudo-spring fracture model is used to predict the crack path of the potential fracture body. The dynamic equations of the SPH particles are solved by using a two-step predictor-corrector scheme with the second-order accuracy, while the Differential Algebraic Equations (DAEs) is solved by using the generalized-alpha method. Finally, a numerical example is presented to validate the effectiveness of the proposed computation methodology. The results indicate that the proposed methodology in the study has the potential to simulate the dynamic fracture phenomenon of practical engineering mechanisms. In the ongoing research, the proposed methodology will be used to study the fracture dynamics of large-scale multibody systems.

### REFERENCES


Dynamic Modeling Approach for a Continuous Moving Belt

Norbert Lorenz∗, Günter Offner#, Martin Sopouch#, Otmar Ertl†

∗ Radon Institute for Computational and Applied Mathematics, Austrian Academy of Science, and MathConsult GmbH, both Altenberger Straße 69, 4040 Linz, Austria
Norbert.Lorenz@mathconsult.co.at

# Advanced Simulation Technologies, AVL List GmbH, Hans-List-Platz 1, 8020 Graz, Austria
[Guenter.Offner, Martin.Sopouch]@avl.com

† MathConsult GmbH, Altenberger Straße 69, 4040 Linz, Austria
Otmar.Ertl@gmail.com

ABSTRACT

This study develops an efficient and realistic modeling approach in order to predict the dynamic response of belt drives. The model assumes a planar configuration of a continuous moving belt, consisting of one belt and an arbitrary number of pulleys. The belt is modeled as a planar viscoelastic element, which considers lateral elastic deformations and its longitudinal movement. For this the equations of motion are derived, in which all effects of inertia, including strain, bending, shear forces, and repulsion forces between pulley and belt are included. The contact between the pulleys and the belt is modeled using a slip-approach formulation with frictional contact governed by a non-linear pulley-belt friction-coefficient characteristic. The dynamic response of the belt drive is studied by incorporating the model into the multi-body software AVL-EXCITE, which can maintain the dynamic simulation of the connected pulley bodies. The verification of the developed model is done for a two-pulley belt drive, where the numerically obtained results of the first and second natural frequency of lateral belt vibrations are compared with the analytic solution. Finally, the applicability of the belt-drive model is demonstrated by simulation of the typical engineering task of a preloaded three-pulley belt drive investigation, considering non-steady operation conditions.

Keywords: belt drives, lateral vibrations, pulley-belt interaction, slip curve, natural frequency.

1 INTRODUCTION

Belt drives are used in numerous applications, such as automotive engines, industrial drives or household applications to transmit power between engines and machines. Because of their simple installation and low maintenance together with their ability to absorb shocks, they are frequently used instead of chain or geared transmission systems. However, they can exhibit complex dynamic behaviors, such as the transverse vibrations of the belt spans, sliding of the belt over the pulley, etc. All these phenomena impact the belt life and also the acoustic comfort. It is therefore of interest to predict the dynamic response of such systems using numerical models. Examples of related simulation models and references to this class of problems can be found in [1], [2] and [3].

In this work a mathematical model for the non-synchronous belt drive is presented, which is able to describe the entire belt (free spans as well as wrapped arc sections). The model makes the need of dissections unnecessary, and has a major interest on the dynamics of the lateral displacements. Furthermore, the belt-pulley interaction is incorporated and modeled by two functions, which define the belt-pulley repulsion force, modeled by elastic contact conditions, and the transferred torque, modeled by a simple slip-approach.
The model can contain an arbitrary layout with any number of connected pulleys, see Figure 1 for a simple two pulley belt drive and a more complex multi-pulley serpentine configuration as used for driving various auxiliaries of a combustion engine. Such drives are termed Front End Auxiliary Drives (FEAD) and represent the most prominent application of non-synchronous belts. In order to simulate the dynamic behavior of the belt, the longitudinal movement and the transversal vibrations are modeled. Therefore the belt is represented as one planar viscoelastic element, which is discretized by a number of points, where the lateral displacement coordinates, according to a reference configuration, are defined.

The contributing forces for the derived equations of motion are the strain forces, the belt bending forces, the repulsion forces between the pulleys and the belt, and the shear forces due to the transferred torque at the contacting pulley-belt surfaces.

The pulleys are modeled as rotating circular bodies. For the dynamic simulation of this contacting pulley elements a flexible multi-body approach is applied. This approach takes 3D linear elastic body dynamics of this structural components into account, cf. [4, 5].

![Figure 1. Examples for non-synchronous belt drive configurations.](image)

This belt drive model is embedded into the flexible multi-body software AVL-EXCITE, which simulates the 3D dynamics of flexible components. Details on the applied multi-body package itself can be found in [6].

2 MATHEMATICAL MODEL

In this study the belt drive, which consists of one belt and an arbitrary number of pulleys, is investigated. The system is divided into the belt and the pulleys, which are modeled separately, and joined together by the contact forces. In the following sections the individual parts of the belt drive model will be described in detail, which includes the contact problem and the investigation of the contact forces, and presents the equations of motion of the whole pulley belt configuration.

2.1 Additional model assumption

For simplification and model reduction the following assumptions are made:

- The variations of the mass density due to belt elongation is ignored.
- Local longitudinal vibrations are neglected. (The global longitudinal elongation, and the local lateral vibrations are considered.)
- The magnitude of the lateral displacement is small in comparison to the radii of the pulleys.
  - Therefore only belt vibrations close to the reference configurations are allowed.
  - Only small translational pulley movements, relative to the initial position, are possible.
2.2 Reference state

The movement of the belt drive can be separated into longitudinal motion and lateral vibration, where the lateral displacement describes the belt configuration as offset from a reference state. This reference configuration agrees to the unloaded ideal geometric shape of the drive. Hence, the reference configuration of the belt can be described by ideal straight segments and circular arcs wrapped around the pulleys, which as a whole describe a closed curve in two dimensions:

\[ \mathbf{x}_0 : [0, L_0] \to \mathbb{R}^2 \quad \text{with} \quad \mathbf{x}_0(0) = \mathbf{x}_0(L_0) \quad \text{and} \quad \| \mathbf{x}_0' \| = 1 \]  

(1)

The parameter is the arc length and \( L_0 \) is the length of the belt on the non-pre-loaded condition, further referred to as idle state or reference configuration. Figure 2 shows such a reference configuration for a simple two pulley belt drive.

![Figure 2. Example of a reference configuration, belt wrapped around two pulleys.](image)

The reference configuration \( \mathbf{x}_0(s) \) and the length \( L_0 \) can be calculated, if the centers \( \mathbf{c}_i(t = 0) \) and the corresponding radii \( r_i \) of all pulleys, \( i \in \{1, ..., N_p\} \), are known. \( N_p \) is the number of pulleys, where \( N_p > 1 \).

To simplify notations, the engagement point \( s_{2i-1} \) and disengagement point \( s_{2i} \) are introduced, where the engagement point is the first point \( s \in [0, L_0] \) where the reference belt comes in contact with pulley \( i \), and the disengagement point is the last point of contact. The length of the contacting segment, on pulley \( i \), is denoted by

\[ l_p^{(i)} = \begin{cases} s_{2i} - s_{2i-1} & s_{2i} \geq s_{2i-1} \\ s_{2i} + L_0 - s_{2i-1} & \text{else} \end{cases} \]  

(2)

The signed curvature of the reference configuration is defined as follows

\[ \kappa(s) := \mathbf{x}_0''(s) R \mathbf{x}_0'(s) \]  

(3)

where \( R \) denotes the 90° counter clockwise rotation matrix:

\[ R := \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \]  

(4)

2.3 Lateral vibrations

The position of a moving belt is distorted with respect to its reference state and can be described by the lateral displacement

\[ w = w(s, t) \]  

(5)

which is a function of arc length (of the reference state) \( s \) and time \( t \). The curve of the belt at time \( t \) is given as:

\[ \mathbf{x}(s, t) = \mathbf{x}_0(s) + w(s, t) R \mathbf{x}_0'(s) \]  

(6)

Therefore the position of the belt at a time point \( t \) can be uniquely defined by (6) based on its reference state (1).
In Figure 3 an example for a displaced belt is shown, where the corresponding reference configuration is depicted in blue.

**Discretization of the lateral movement**

The lateral displacement $w$ of the belt is represented as a linear combination of $N$ time invariant basis functions $w_i \quad i \in \{1, 2, \ldots, N\}$:

$$w(s, t) = \sum_{i=1}^{N} q_i(t) w_i(s) = q(t)^T w(s) \quad (7)$$

**2.4 Longitudinal elongation, and movement**

According to Section 2.1 the local longitudinal vibrations are neglected, but the global longitudinal elongation is considered. This longitudinal elongation is caused due to the lateral vibrations, and due to the moving of the pulleys respectively. These effects are already modeled by the lateral state variable $w(s, t)$ introduced in the last Section 2.3.

Additionally, the global longitudinal movement of the belt is modeled by a global uniform displacement variable $q_{long}$, where $v$ denotes the longitudinal belt velocity.

**2.5 Equation of motion**

The Lagrange-d’Alembert principle, cf. [7], is used in order to derive the equation of motion. The state variables for this problem are the lateral displacements and the longitudinal displacements. The contributing forces are the pulley-belt contact forces, the strain forces due to longitudinal stiffness, the shear forces between pulley and belt and the bending forces. Additionally, a simple global slip approach is incorporated to model the variable friction coefficient at the pulley belt contact zone.

**2.5.1 Kinetic energy**

The kinetic energy is given as:

$$T = \frac{m_l}{2} \int_0^{L_0} ||\vec{v}||^2 ds = \frac{m_l}{2} \int_0^{L_0} \frac{d}{dt} \tilde{x} \left(s + \int_0^t v \, dt \right) \, ds = \frac{m_l}{2} \int_0^{L_0} ||\vec{x}' + \tilde{x}||^2 \, ds \approx \frac{m_l}{2} \int_0^{L_0} \left( (\dot{w} + w')^2 + v^2 (1 - w \kappa)^2 \right) ds \quad (9)$$

$m_l$ is the mass density of the belt (using $w^2 \kappa^2 \approx 0$ according to Section 2.1).

**2.5.2 Belt-pulley repulsion**

The belt pulley interaction with respect to radial (=normal) traction forces can be described by a function $F_{rep}(d_i, \dot{d}_i)$, which gives the repulsion force as a function of the depth $d_i$ and rate $\dot{d}_i$ of
indentation for pulley $i$. 

$$F_{rep}^{(i)}(d_i, \dot{d}_i) = \begin{cases} d_i E_{emb}^{(i)} + d_i D_{emb}^{(i)} & d_i > 0 \\ 0 & d_i \leq 0 \end{cases}$$ (10)

where $E_{emb}^{(i)}$ is the modulus of embedding and $D_{emb}^{(i)}$ is a corresponding parameter to incorporate damping. Assuming perfect circular shaped pulleys with center $\vec{c}_i$ and radius $r_i$, the repulsion force vector acting on the beam at point $s$ is given by:

$$\vec{F}_{rep}^{(i)}(s) = F_{rep}^{(i)}(d_i, \dot{d}_i) \frac{\vec{x} - \vec{c}_i}{\|\vec{x} - \vec{c}_i\|}$$ (11)

Hence, the virtual work due to repulsion is

$$\delta W_{rep} = -\sum_{i=1}^{N_p} \int_0^{L_0} \vec{F}_{rep}^{(i)}(s) \delta \vec{x} ds$$ (12)

where the virtual displacement is $\delta \vec{x} = \vec{x}_0 \delta q_{long} + R \vec{v}_0 \delta w$.

### 2.5.3 Strain

The virtual work due to longitudinal strain is

$$\delta W_{strain} = -\int_0^{L_0} F_{long}(s,t)(w'' + \kappa) \delta w ds$$ (13)

**Note 1:** Equation (13) defines the virtual work due to lateral movement of the belt against the retaining action of the longitudinal forces. The term $(w'' + \kappa)$ denotes the relationship between the lateral and longitudinal direction in case that the belt is laterally displaced from its reference condition.

Here $F_{long}$ is the longitudinal force, which is

$$F_{long}(s,t) = F_0 + \frac{E A}{L_0} \Delta L + D A \frac{\Delta L}{L_0} + \sum_{i=1}^{N_p} F_{torque}^{(i)}(s,t)$$ (14)

$F_0$ is an initial longitudinal preload and $\Delta L$ is the actual total elongation of the belt. $E A$ and $D A$ are the stiffness and damping coefficient due to its longitudinal elongation. $F_{torque}^{(i)}(s,t)$ is the additional longitudinal force introduced by pulley $i$, which acts on the belt with a torque $M_{p}^{(i)}(t)$. In order to estimate the shape of the function $F_{torque}^{(i)}(s,t)$ from the applied torque, it is assumed that the shear force $F_{shear}^{(i)}$, which finally leads to the torque transfer, is constant for the belt segment that is wrapped around pulley $i$.

$$F_{shear}^{(i)}(s,t) = \frac{\kappa_i M_{p}^{(i)}(t)}{A_{p}}$$ (15)

**Note 2:** Due to the assumption that the longitudinal speed is the same for the entire belt, also the longitudinal acceleration must be equal everywhere. Hence, the acceleration of the belt due to applied forces,

$$\frac{dF_{long}}{ds}(s,t) + \sum_{i=1}^{N_p} F_{shear}^{(i)}(s,t) = \sum_{i=1}^{N_p} \left( \frac{dF_{torque}}{ds}(s,t) + F_{shear}^{(i)}(s,t) \right)$$ (16)

must be independent of $s$. 

324
To ensure the assumptions in 2.1, also described in Note 2, the following setting is used:

\[ F_{\text{torque}}^{(i)}(s,t) = M_p^{(i)}(t) \tau_i(s) \]  

(17)

with

\[ \tau_i(s) = \frac{1}{2} k_i(1 - l_p^{(i)}) \left\{ \begin{array}{ll}
1 + \frac{s - s_{2i} - 2s}{l_p^{(i)}} & s \in [s_{2i-1}, s_{2i}] \\
1 - \frac{s - s_{2i-1} - s_{2i}}{L_0 - l_p^{(i)}} & s \in [s_{2i}, s_{2i-1} + L_0]
\end{array} \right. \]  

(18)

### 2.5.4 Shear forces

In the previous section the shear forces between the pulleys and belt have been incorporated for the longitudinal strain. These shear forces also lead to longitudinal acceleration. The corresponding virtual work is given as

\[ \delta W_{\text{shear}} = \delta q_{\text{long}} \sum_{i=1}^{N_p} \int_0^{L_0} F_{\text{shear}}^{(i)}(s,t) \, ds = \delta q_{\text{long}} \sum_{i=1}^{N_p} k_i M_p^{(i)}(t) \]  

(19)

### 2.5.5 Bending

The virtual work due to bending is (derived from the bending energy, cf. [8] and Section 2.1)

\[ \delta W_{\text{bend}} = \int_0^{L_0} \delta w''(EI(\kappa + w'') + D_I w'') - \delta w'' D_I v(\kappa + w'') \, ds \]  

(20)

\( EI \) and \( DI \) are stiffness and damping coefficients due to bending of the belt.

### 2.5.6 Equations of motion

The equation of motions can be obtained by

- using the variation principal of d’Alembert, cf. [7], from:

\[ \delta T = \delta W_{\text{rep}} + \delta W_{\text{strain}} + \delta W_{\text{bend}} + \delta W_{\text{shear}} \]  

(21)

- incorporating of the assumptions from Chapter 2.1,
- collecting the corresponding expressions for the coefficients \( \delta q \) and \( \delta q_{\text{long}} \),
- and some algebraic modifications.

Finally the state of the belt drive can be expressed by a system of equations for the transversal motion:

\[
\sum_{i=1}^{N_p} M_p^{(i)}(t) (\vec{q}_i + T_0 \vec{q} + 2C_i \dot{q} + (K_0 - C_i q) \vec{q}) - (F_0 + EA \frac{\Delta L}{L_0} + DA \frac{\Delta L}{L_0}) (K_0 - C_i q) \\
+ \left( EI(K_2 - C_i^2 q) + D_IC_i \dot{q} + D_Iv(C_2 q - K_3) \right)
\]

\[ = \sum_{i=1}^{N_p} M_p^{(i)}(t) (\vec{q}_i + T_0 \vec{q}) + \int_0^{L_0} w F_{\text{rep}}^{(i)}(d_i, \dot{d}_i) (\vec{x} - \vec{c}_i) R \vec{q} \\
joint \text{force contribution of pulley } i
\]

and a single equation for the belts global longitudinal motion:

\[
m_l \left[ \ddot{L}_0 \dot{v} - 2v \dot{q} T (K_0 - C_2 q) + \dot{q} T C_1 q \right] = \sum_{i=1}^{N_p} k_i M_p^{(i)}(t) + \int_0^{L_0} F_{\text{rep}}^{(i)}(d_i, \dot{d}_i) (\vec{x} - \vec{c}_i) N_0 \vec{q} \\
joint \text{force contribution of pulley } i
\]  

(22)
The Matrices and Vectors $C_0 - C_3$, $T_i$, $K_0 - K_3$, $R_n$ are model parameters depending on the reference geometry and the used ansatz functions, therefore they can be calculated initially by:

$$
K_0 := \int_0^{L_0} \kappa w \, ds, \quad K_2 := \int_0^{L_0} \kappa w' \, ds, \quad K_3 := \int_0^{L_0} \kappa w'' \, ds,
$$

$$
R_n := \int_0^{L_0} \tau_n(s) \kappa w \, ds \quad 1 \leq n \leq N_p,
$$

and matrices by

$$
C_0 := \frac{\kappa}{2} \int_0^{L_0} w'' w' \, ds, \quad C_1 := - \frac{\kappa}{2} \int_0^{L_0} w' w'' \, ds, \quad C_2 := - \frac{\kappa}{2} \int_0^{L_0} w'' w'' \, ds,
$$

$$
C_3 := \frac{\kappa}{2} \int_0^{L_0} w'' w'' \, ds, \quad C_4 := - \frac{\kappa}{2} \int_0^{L_0} w'' w'' \, ds,
$$

$$
T_i := \int_0^{L_0} \tau_n(s) w w'' \, ds \quad 1 \leq n \leq N_p.
$$

### 2.5.7 Forces and moments of belt on pulley

By definition and due to Newton’s third law the torque, which acts on pulley $i$, is equal to

$$
M_{belt\rightarrow pulley}^{(i)}(t) = -M_p^{(i)}(t)
$$

The force on the pulley is given as:

$$
F_{belt\rightarrow pulley}^{(i)} = \int_0^{L_0} F_{rep}(s) \, ds
$$

### 2.5.8 Torque transfer between pulley and belt

As outlined in 2.5.3 a constant distribution of shear forces across the arc of contact between belt and pulley is assumed. The corresponding torque $M_p^{(i)}(t)$ is computed by using a classical slippage approach. The approach is based on a variable friction coefficient $\mu(i)$, which depends on the average relative tangential velocity (referred to as slip $\dot{s}^{(i)}$) in the contact between belt and pulley $i$:

$$
\mu(i) = \begin{cases} 
\mu_{max} \sin(\pi - \arcsin(\mu_{0}^{(i)})) & \dot{s}^{(i)} = \infty \\
\mu_{max} \sin\left(2 - \frac{2}{3} \arcsin(\mu_{0}^{(i)}) \right) \arctan\left(\frac{\dot{s}^{(i)}}{\mu_{max}}\right) & \text{otherwise}
\end{cases}
$$

Equation (28) defines a so called slip curve as depicted in Figure 4, parameters $\mu_{max}^{(i)}$ (=maximum coefficient of friction), $\dot{s}_{\mu_{max}}^{(i)}$ (=percentage of slip at maximum coefficient of friction) as well as $\mu_{f\infty}^{(i)}$ (=percentage of $\mu_{max}$ at full slip) are used as inputs.

$\dot{s}^{(i)}$ denotes the slip value, which gives a 'normalized measure' of the relative circumferential velocity arising at the belt pulley interface:

$$
\dot{s}^{(i)} = \frac{v_{belt} - v_{pulley}^{(i)}}{\max(|v_{belt}|, |v_{pulley}^{(i)})}
$$

Herein $v_{belt}$ agrees to the belts transport velocity $v$, and $v_{pulley}^{(i)}$ denotes the tangential velocity $v_{pulley}^{(i)} = r_i \omega_{p}^{(i)}$, where $\omega_{p}^{(i)}$ is the rotational velocity of pulley $i$. Finally the transferred torque is computed according to Coulomb’s law:

$$
M_p^{(i)} = \mu(i) \dot{s}^{(i)} F_n^{(i)} r_i = \mu(i) \| F_{belt\rightarrow pulley}^{(i)} \| r_i
$$

Bearing in mind the emphasis on lateral belt vibrations the actual model is based on a simplified constant shear force distribution throughout the contact arc. For other more advanced contact models, which also employ creep-rate dependent friction laws and differentiate between adhesive and active arc, see also e.g. [2], [9] or [10].
2.6 Basis functions

Since the Hermite cubic base functions (splines) are very popular for this type of problems [11], they will also be used here. They are two-times differentiable and matrices (25) and vectors (24), even for the included third derivatives, can be calculated. Furthermore, all these matrices are band matrices and the effort for the corresponding matrix-vector multiplications scale linearly with the number of basis functions. The basis functions are defined as

\begin{align*}
    w_{2j+1}(s) := (x_i - x_{i-1}) h_{j1} \left( \frac{s + x_{i-1}}{x_i - x_{i-1}} \right) + (x_{i+1} - x_i) h_{j0} \left( \frac{s + x_i}{x_{i+1} - x_i} \right)
\end{align*}

\[j \in \{1, 2\}, \quad i \in \{0, 1, \ldots, N_{\text{node}}\}\tag{31}\]

where \(x_0, \ldots, x_{N_{\text{node}}}\) are the nodes, for which

\[0 = x_0 < x_1 < \ldots < x_{N_{\text{node}}} = L_0\tag{32}\]

holds. \(N_{\text{node}}\) is the number of nodes and \(x_0\) and \(x_{N_{\text{node}}}\) correspond to the same node due to periodicity. Using the function

\begin{align*}
    h_{00}(t) := (2t^3 - 3t^2 + 1) \theta(t) \theta(1-t) \\
    h_{10}(t) := (t^3 - 2t^2 + t) \theta(t) \theta(1-t) \\
    h_{01}(t) := (-2t^3 + 3t^2) \theta(t) \theta(1-t) \\
    h_{11}(t) := (t^3 - t^2) \theta(t) \theta(1-t)
\end{align*}

\[\theta\] denotes the Heaviside step function with the convention \(\theta(0) := \frac{1}{2}\).

Since there are two basis functions per node, there are \(N = 2N_{\text{node}}\) basis functions, see (7) for the corresponding representation of the lateral displacement function \(w(s, t)\).

3 NUMERICAL RESULTS

This belt drive model is embedded into the flexible multi-body software AVL-EXCITE (cf. [6]), which provides the dynamic simulation of the corresponding pulley bodies, which are joined by the derived contact forces and moments.

3.1 Step function excitation

The verification of this developed model is done for a two-pulley belt-drive, similar to Figure 2, where the numerically obtained results of the lateral vibrations are compared with an analytic solution, cf. [12]. The parameters of the configuration are given in Table 1.
Table 1. Belt drive parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Radius of the driver and the driven pulley</td>
<td>$r_1$, $r_2$</td>
<td>0.25 m</td>
</tr>
<tr>
<td>Distance of pulley centers</td>
<td>$d$</td>
<td>0.5 m</td>
</tr>
<tr>
<td>Belt mass, per unit length</td>
<td>$m_l$</td>
<td>0.096 kg/m</td>
</tr>
<tr>
<td>Longitudinal stiffness</td>
<td>$E_A$</td>
<td>182400 N</td>
</tr>
<tr>
<td>Longitudinal damping</td>
<td>$D_A$</td>
<td>18.24 Ns</td>
</tr>
<tr>
<td>Bending rigidity</td>
<td>$E I$</td>
<td>0.0312 Nm$^2$</td>
</tr>
<tr>
<td>Bending damping</td>
<td>$D I$</td>
<td>3.12E-5 Ns$^2$m$^2$</td>
</tr>
<tr>
<td>Preload</td>
<td>$F_0$</td>
<td>500 N</td>
</tr>
<tr>
<td>Embedding stiffness into the pulleys</td>
<td>$E_{emb}$</td>
<td>100.0E6 N/m$^2$</td>
</tr>
<tr>
<td>Embedding damping coefficient</td>
<td>$D_{emb}$</td>
<td>0.1E-5 Ns$^2$m$^2$</td>
</tr>
<tr>
<td>Frictional slip coefficient $i = 1, 2$</td>
<td>$\mu_{max}^{(i)}$</td>
<td>1 -</td>
</tr>
<tr>
<td>Frictional slip coefficient $i = 1, 2$</td>
<td>$\bar{s}<em>{\mu</em>{max}}^{(i)}$</td>
<td>10 %</td>
</tr>
<tr>
<td>Frictional slip coefficient $i = 1, 2$</td>
<td>$\mu_{\bar{s甜}}^{(i)}$</td>
<td>50 %</td>
</tr>
</tbody>
</table>

In this section a lateral excitation on the belt configuration is performed, where the following settings is used:

- The step function is performed by a 20 mm vertical displacement of pulley center 2 (shown in Figure 5), the corresponding pulley positions are shown in Figure 6.
- Drive is at standstill -> no rotation is applied.
- No load torque applied -> no power transmission.

Figure 5. Step function.

Figure 6. Positions of the belt configuration, before and after the performed step.

Figure 7 shows the results of the upper span of the belt when performing this pulley movement numerically. Sub-figure (a) present the lateral vibration during the simulation time, where Sub-figure (b) show the corresponding belt tension course. The reduced belt tension of approximately 400 N at the beginning (see Figure 7 (b)) in comparison to the setting of 500 N (see Table 1) results due to the considered embedding of the belt into the pulleys.

According to [13] the square of the $n$-th transverse natural frequency $\tilde{\omega}_n^2$ of a simply supported, stationary, tensioned beam, which also includes bending, is a linear function of the tension $T$:

$$\tilde{\omega}_n^2 = E I \frac{n^4 \pi^4}{m_l t^4} + T \frac{n^2 \pi^2}{m_l t^2}$$

(34)
where \( l \) is the length of the observed free span.

Therefore it’s possible to calculate the theoretical natural frequency in dependence on the belt tensions, and compare them to the observed oscillations. At the beginning (0-1 s calculation time) the belt tension is approximately at 400 N, which leads according to equation (34) to a first natural frequency of 64.55 Hz and a second of 129.90 Hz. At the end (1.1-2 s calculation time) the belt tension is approximately 535 N, which leads to a first natural frequency of 74.73 Hz, and a second of 149.90 Hz. Figure 8 and Figure 9 show the numerically obtained first and second natural frequency detected by a fast Fourier transformation (FFT) of the lateral vibrations before and after the performed step. Comparing them to the calculated values show only very small differences. Table 2 and Table 3 show a summary of these results before and after the performed step.

### 3.2 Application example

Finally, the applicability of the belt-drive model is demonstrated by means of a common preloaded three-pulley layout, considering non-steady operation conditions. The transmission consists of a driving pulley interconnected to an electric motor as well as a driven pulley attached to a centrifugal pump (cf. Figure 10). The third pulley represents a so called fixed tensioner idler. It is used to apply the required belt-preload by moving it into the belt span until the desired tension is developed. This tensioning process is part of the installation procedure, during actual operation the

![Figure 7. Lateral displacement and tension of upper span (at 1/2 span length)](attachment:image)

![Figure 8. Fast Fourier transformation (FFT) of latereral vibrations of upper span (1/2 span length).](attachment:image)

![Figure 9. Fast Fourier transformation (FFT) of lateraler vibrations of upper span (1/2 span length).](attachment:image)
tensioner is fixed. In the simulation model the driving pulley has a kinematic boundary condition and is ramped up from standstill to a rotary speed of 3000 rpm within a startup time of 3 s. Additionally there is an offset between driver pulley’s center and the rotation axis of 0.1 mm, which considers a radial runout of this pulley. This eccentricity serves as an excitation source providing perturbations with the 1st order of the corresponding pulley rotation and is used to promote transverse belt vibrations. Moreover, at the driven pulley, which has inertia of 0.001 kgm$^2$, a constant load torque of 50 Nm is applied. The main emphasis of this investigation is on the lateral belt vibrations evolving in the tight and loose spans during the drive’s startup phase. The parameters of the configuration are given in Table 4.

Table 2. Step function excitation results of the pre-step interval.

<table>
<thead>
<tr>
<th></th>
<th>first natural frequency</th>
<th>second natural frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>analytic result</td>
<td>64.55 Hz</td>
<td>129.90 Hz</td>
</tr>
<tr>
<td>numerical result (cf. Figure 8, 9)</td>
<td>65.00 Hz</td>
<td>131.19 Hz</td>
</tr>
</tbody>
</table>

Table 3. Step function excitation results of the post-step interval.

<table>
<thead>
<tr>
<th></th>
<th>first natural frequency</th>
<th>second natural frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>analytic result</td>
<td>74.73 Hz</td>
<td>149.90 Hz</td>
</tr>
<tr>
<td>numerical result (cf. Figure 8, 9)</td>
<td>75.42 Hz</td>
<td>153.12 Hz</td>
</tr>
</tbody>
</table>

Figure 10. Application of a belt drive.

Figure 11 shows the velocity course of the run up simulation, where the black line indicates the tangential velocity of the driving pulley (electric motor), the red dashed line the tangential velocity of the driven pulley (centrifugal pump), and the blue dot-dashed line the longitudinal velocity of the belt. According to the applied creep-rate dependent friction law, cf. Section 2.5.8, the slip-approach reveals small differences in the tangential velocities in order to transmit the required torque. Slight fluctuations visible in the velocity traces are caused by the eccentric pulley rotation.
Table 4. Belt drive parameters of the application example.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Belt mass, per unit length</td>
<td>$m_l$</td>
<td>0.095 kg</td>
</tr>
<tr>
<td>Longitudinal stiffness</td>
<td>$EA$</td>
<td>240000 N</td>
</tr>
<tr>
<td>Longitudinal damping</td>
<td>$DA$</td>
<td>20.0 Ns</td>
</tr>
<tr>
<td>Bending rigidity</td>
<td>$EI$</td>
<td>0.025 Nm$^2$</td>
</tr>
<tr>
<td>Bending damping</td>
<td>$DI$</td>
<td>0.025E-4 Nm$^2$</td>
</tr>
<tr>
<td>Preload</td>
<td>$F_0$</td>
<td>300 N</td>
</tr>
<tr>
<td>Embedding stiffness into the pulleys</td>
<td>$E_{emb}$</td>
<td>140.0E6 Nm$^2$</td>
</tr>
<tr>
<td>Embedding damping coefficient</td>
<td>$D_{emb}$</td>
<td>0.1E-5 Ns m$^2$</td>
</tr>
<tr>
<td>Frictional slip coefficient $i = 1, 2, 3$</td>
<td>$\mu_{(i)}^{\text{max}}$</td>
<td>1.5 -</td>
</tr>
<tr>
<td>Frictional slip coefficient $i = 1, 2, 3$</td>
<td>$\bar{\mu}_{(i)}^{\text{max}}$</td>
<td>2 %</td>
</tr>
<tr>
<td>Frictional slip coefficient $i = 1, 2, 3$</td>
<td>$\bar{\mu}_{(i)}^{\text{max}}$</td>
<td>80 %</td>
</tr>
</tbody>
</table>

Figure 11. Tangential velocities of driver pulley, driven pulley and longitudinal belt velocity.

Figure 12 shows the lateral vibrations of the free belt spans, each in the middle of the corresponding span lengths. While during the initial phase of the run-up the amplitudes are just of minor extend (<2 mm), significant vibrations are excited after 2 s in the tight span and after 1.2 s in the both short slack spans. In particular the lower left slack span between the disengagement zone at the driving pulley and the tensioner idler is exposed to considerable lateral movements up to ±10 mm. For the upper (=tight) span at 2 s the excitation frequency resulting from the pulley eccentricity is at 2000 rpm $\approx$ 33 Hz, which matches the area of the so called 3rd parametric instability, cf. [13], with a center frequency of 2/3 of the natural transverse frequency ($\tilde{\omega}_1$) of the corresponding span at the apparent belt tension and belt speed. In contrast for the short lower spans the increase of lateral movement can be traced back to a match between a higher order harmonics of the runout excitation with the classical primary instability region given by $2\tilde{\omega}_1$.

4 SUMMARY, CONCLUSIONS AND OUTLOOK

A model for the dynamic simulation of a continuous moving belt with a particular focus on lateral span vibrations is discussed. For this purpose a physical approach, which includes belt’s inertia effects, longitudinal and bending stiffness, shear forces, as well as pulley-belt repulsion, is derived. For the torque transfer between pulley and belt a simple global slip-approach is applied. The validations of the predicted lateral vibrations are done with a two pulley belt drive. The numerically obtained results in terms of natural frequency of the lateral vibrations are compared with the analytic solution. Finally, the applicability of the belt-drive model is demonstrated by simulation of
Figure 12. Lateral vibrations of upper and lower spans (1/2 span length).

a popular preloaded three pulley belt drive design, considering non-steady operation conditions. The following conclusion can be drawn:

- The natural frequencies of the predicted lateral vibrations show reasonable comparison to the analytically calculated values. Moreover, vibrations observed under the parametric excitation case are in agreement to the expected instability areas.

- Compared to a discrete belt model the continuous approach shows significant advantages with respect to computational expense.

- Since the belt is modeled as a whole, excitation arising from belt-to-pulley interactions in the engagement/disengagement zones are incorporated efficiently without the need of additional coupling conditions between free and wrapped belt sections.

- However, currently there are two drawbacks in the applied model formulation:
  - Large global movements of the pulleys are not possible.
  - Local longitudinal vibrations are not considered, which is a pre-condition for a future advanced belt-pulley contact formulation.

5 ACKNOWLEDGMENTS

A part of this work has been supported by the Upper Austrian Government within the framework "Innovatives Oberösterreich 2010plus".
REFERENCES


Efficient Formation and Solution of Equations of Motion for Multibody Systems with Bodies of Mixed Definition Using DCA

Jeremy J. Laflin∗, Kurt S. Anderson∗, Imad M. Khan#

∗Rensselaer Polytechnic Institute
110 8th Street, Troy, NY, USA
laflij@rpi.edu, anderk5@rpi.edu

#Ford Motor Company
Dearborn, MI, USA
ikhan21@ford.com

ABSTRACT
Two popular methods, Absolute Nodal Coordinate Formulation (ANCF) and Geometrically Exact Beam Formulation (GEBF) that describe the large-deformations, translations, and rotations of highly-flexible bodies are examined for integration into an adaptive Divide-and-Conquer Algorithm framework. Although both of these methods can separately be used in the DCA-framework, only the GEBF is suitable for systems that also contain rigid-bodies or flexible-bodies modeled with an FFR formulation (small deformation). Furthermore, adaptive changes in model definition are possible using the GEBF due to the generalized-coordinates used. These reasons are discussed and the equations-of-motion using GEBF in the DCA-framework are derived.

1 INTRODUCTION
Although there has been significant progress associated with modeling and simulating dynamic systems in an adaptive manner [13, 17, 23–26], which ideally considers only those degrees-of-freedom that are deemed important, there are still opportunities for significant further computational savings. Ideally, an adaptive simulation of complex systems should self-identify and utilize different model types which are best suited to the nature of the local behavior of each subdomain. As such, a complex system may ultimately be comprised of rigid bodies (no deformation), flexible bodies (small deformation), and highly-flexible bodies (large deformations). A decrease in computational labor may be achieved by adjusting the definition of the computational model to best match the needs of the associated subdomain, which results in fewer degrees-of-freedom without loss of the accuracy of predicting the system’s behavior.

This is particularly of interest for systems spanning a variety of scales. These adaptive techniques can be used to transition resolution of one region of the model to a different scale so that the computational burden is reduced or the model fidelity of the region is increased. Due to the large number of degrees-of-freedom, biomolecular systems necessitate coarse-graining of the fully atomistic system and are well suited to make use of adaptive changes in type and location of degrees-of-freedom. Figure 1 is an example of the type of system that can only be simulated on premier computing facilities (without coarse graining) and stands to reap a significant decrease in simulation time by making use of these adaptive techniques. By intelligently coarse-graining the system, rigid and flexible bodies can accurately reproduce the overall system behavior with far fewer degrees-of-freedom. These various body types could be used to represent various scales of interest for the HIV-capsid shown in Fig. 1 where a particular residue may be modeled as a collection of atoms or a rigid-body, or the full hexamer-chain may be modeled as a flexible-body undergoing large deformations. For such complex systems, it is impossible to know a priori which body definition should be used so that the behavior of various regions of the model reproduced at the appropriate scale. Therefore, adaptive changes in body-definition are necessary in addition to changes in the number and location of the degrees-of-freedom.

Aggregating various bodies into a body (subdomain) of a new type is a matter of monitoring various degrees-of-freedom, making a determination of which degrees-of-freedom to add or remove and changing the resolution of the model to reflect the desired change. The decision to add or
Figure 1: HIV-Capsid (A) as an example of a multi-scale biopolymeric system consisting of a capsid-hexamer (B) made up of many hexamer-chains (C) which is composed of many residues (D).

remove various degrees-of-freedom may be based on a variety of indicators including, but not limited to, statistics collected from the degrees-of-freedom, physics-based metrics, or knowledge-based metrics.

It has been demonstrated that the Divide-and-Conquer Algorithm (DCA) can form and solve the equations-of-motion for the state derivatives associated with a system comprised of rigid-bodies and flexible bodies undergoing small deformations. Additionally, the DCA has been used to form and solve the equations-of-motion for systems of highly-flexible bodies where the Absolute Nodal Coordinate Formulation (ANCF) has been used to spatially model the large nonlinear deformation of the body [11]. However, the DCA cannot readily form and solve the equations-of-motion for systems comprised of rigid or flexible, and highly-flexible bodies where the ANCF is used to model the large deformations. This is because the ANCF formulation uses global slopes as generalized coordinates instead of rotation coordinates and is largely incompatible with the rotation coordinates generally used with rigid-bodies and flexible bodies undergoing small deformations that are modeled with a Floating Frame of Reference (FFR) formulation, which is used in most multibody formulations.

This places a restriction on the type of systems simulated with the adaptive DCA framework; highly-flexible bodies must be heavily substructured and a method suitable for small deformations, such as the Floating Frame of Reference (FFR), must be used to model the flexibility of each substructure. This results in an undesirable increase in the number of bodies in the system and a compounding increase in the number of degrees-of-freedom. Alternatively, special computational tools would have to be put in place to transform the state variables used with ANCF to
those compatible with rigid and flexible bodies. Furthermore, implementing such tools to facilitate adaptive changes in model fidelity and definition may add significant computational burden. Therefore, the Geometrically Exact Beam Formulation (GEBF) is used to model highly-flexible bodies in the current adaptive DCA framework due to this method’s ability to correctly handle large nonlinear flexible-body deformations, while still using rotational coordinates that can be integrated more easily into the current framework.

1.1 The DCA for Forming and Solving Equations of Motion for Multibody Systems

The DCA was first introduced by Featherstone for both open-loop [6] and closed-loop [7] topologies and increases the efficiency with which the equations-of-motion can be formed and solved numerically. Additionally, there have been a number of modifications to the original method [3–5, 14–16, 19–22]. The basic method is reproduced herein using the notation of Mukherjee and Anderson [18].

![Figure 2: Assembled Body](image)

The DCA consists of two recursive processes: assembly and disassembly. These recursive processes take place using a hierarchical tree structure. The tree structure is defined by the kinematic joints connecting the bodies of the system. Typically the inboard and outboard joint of the body coincides with a reference point called a handle. The inverse inertial properties of two adjoining parent bodies are combined to represent a fictitious assembly (child body), see figure. 2. This is possible because the kinematics of the joint are known, which allows the constraint forces acting at the connecting joint to be excluded from the equations-of-motion of the outboard handles of the parent bodies (the handles of the child body). The relative motion between the parent bodies is captured by an equation that describes the amount of motion happening in the directions of motion that are allowed by the joint, which are known. This process is repeated until there is only one body, the root body (see Fig. 3). At this point the boundary conditions are known and the equations-of-motion can be solved, which do not contain any of the constraint forces acting at the non-terminal joints.

In general, a body of the system may possess any number of handles, though the basic aspects of the method are most easily conveyed by discussing a chain system. For such systems, each body possesses two handles that connect the body to its inboard and outboard neighbors. The basic method involves writing the spatial equations-of-motion corresponding to these two reference
Figure 3: Assembly and Disassembly Process for Binary Tree Configuration

points (handles) $H_i$, $i = 1, 2$ on each body, as

$$A_1^i = \zeta_{11}^k F_{1c}^k + \zeta_{12}^k F_{2c}^k + \zeta_{13}^k,$$

(1)

$$A_2^i = \zeta_{21}^k F_{1c}^k + \zeta_{22}^k F_{2c}^k + \zeta_{23}^k,$$

(2)

and

$$A_{1+1}^i = \zeta_{11}^{k+1} F_{1c}^{k+1} + \zeta_{12}^{k+1} F_{2c}^{k+1} + \zeta_{13}^{k+1},$$

(3)

$$A_{2+1}^i = \zeta_{21}^{k+1} F_{1c}^{k+1} + \zeta_{22}^{k+1} F_{2c}^{k+1} + \zeta_{23}^{k+1}.$$

(4)

These bodies, Body$^i$ and Body$^{i+1}$, connected by a kinematic joint $j$ and therefore are subject to the kinematic constraint

$$P^j \dot{u} = A_1^{i+1} - A_2^i - P^j u.$$  

(5)

$A_1^i$ and $F_1^i$, are the spatial acceleration of, and force on, handle $i$ respectively and are defined as

$$A_1^i = \begin{bmatrix} \ddot{\alpha}_1^i \\ \ddot{\alpha}_2^i \end{bmatrix},$$

(6)

and

$$F_1^i = \begin{bmatrix} \tau_1^i \\ f_1^i \end{bmatrix}.$$  

(7)

The rotational acceleration of Body$^i$ is $\ddot{\alpha}_1^i$, $\ddot{\alpha}_2^i$ is the translational acceleration of the reference point $H_i$, and $\tau_1^i$ and $f_1^i$ are the constraint torques and forces acting at $H_i$ respectively. The $\zeta_{ij}^k$ ($i, j = 1, 2$) terms are the spatial matrix representations of the inverse inertial properties at the handles, while $\zeta_{ij}^k$ ($i = 1, 2$) contains applied forces acting on the body and other velocity dependent terms. The resulting set of equations, Eqs. (1 - 4), can be reduced by exploiting the fact that the constraint forces are equal and opposite, i.e., $F_{1c}^k = -F_{2c}^{k+1}$, and that the kinematics of the connecting joint are specified. Specifically, Eqn. (5) describes the relative acceleration between connecting bodies using the generalized acceleration $\ddot{u}$ along known directions defined by the connecting joint partial velocity (mode of motion) $P^j$. The equations-of-motion for the assembled fictitious pseudo-body Body$^{i,k+1}$, at $H_1^i$ and $H_2^{i+1}$ can be expressed as

$$A_1^i = \frac{r_{11}}{5} F_{1c}^k + \frac{r_{12}}{5} F_{2c}^k + \frac{r_{13}}{5} + \frac{r_{14}}{5} F_{1c}^{k+1} + \frac{r_{15}}{5} F_{2c}^{k+1},$$

(8)

and

$$A_{1+1}^i = \frac{r_{11}}{5} F_{1c}^k + \frac{r_{12}}{5} F_{2c}^k + \frac{r_{13}}{5} + \frac{r_{14}}{5} F_{1c}^{k+1} + \frac{r_{15}}{5} F_{2c}^{k+1}. $$

(9)
by algebraically eliminating the constraint forces at the connecting joint. The resulting equations (8 & 9) are of the same form as the equations-of-motion for the handles of any generic body.

In the above equations, \( \zeta_k^{ij} \) represents the inertial quantities of the fictitious pseudo-body resulting from the assembly of Body\( ^k \) and Body\( ^{k+1} \). For the derivation of the inverse inertial terms and the details of the assembly process, the reader is referred to the work of Featherstone [6] or Mukherjee and Anderson [18]. This assembly process is then repeated recursively, until only a single assembled pseudo-body remains (root body), as shown in Fig. 3. This is possible because the form of the equations-of-motion for the handles of an assembled body is indistinguishable from the form of the equations-of-motion for the handles of a generic body. The assembly process yields the equations-of-motion associated with the two boundary handles

\[
A_1 = \frac{\zeta_1^{1a} F_1^{1a}}{\zeta_1^{1a}} + \frac{\zeta_1^{1b} F_2^{1b}}{\zeta_1^{1b}} + \frac{\zeta_1^{1c} F_3^{1c}}{\zeta_1^{1c}}
\]

and

\[
A_2 = \frac{\zeta_2^{1a} F_1^{1a}}{\zeta_2^{1a}} + \frac{\zeta_2^{1b} F_2^{1b}}{\zeta_2^{1b}} + \frac{\zeta_2^{1c} F_3^{1c}}{\zeta_2^{1c}}
\]

which are written in terms of only the spatial inertial quantities of all bodies in the system and the constraint forces acting at the two handles of the root body (boundary handles).

The spatial accelerations of, and constraint forces acting at \( H_1^1 \) and \( H_2^2 \) can now be determined using the known boundary conditions. After determining these quantities, the disassembly process begins, in which all unknown spatial accelerations of the handles and constraint forces acting at all connecting joints are determined. This recursive process determines the constraint forces acting at a joint in terms of the constraint forces acting at the handles of the assembly, and the inertial properties of the assembled body, as

\[
F_{1c}^{k+1} = W \frac{\zeta_2^{1a} F_1^{1a}}{\zeta_2^{1a}} - W \frac{\zeta_2^{1b} F_2^{1b}}{\zeta_2^{1b}} + Y
\]

The terms \( W \) and \( Y \) are terms containing inertial properties from the assembly of the two bodies, see Featherstone [6], or Mukherjee and Anderson [18] for derivation of these terms. Once this constraint force acting at a joint is determined, the spatial accelerations of the handles that are connected by this joint can be determined using Eqns. (8 & 9). This allows the computation of the generalized acceleration (\( \dot{\mu} \)) at the joint using Eqn. (5).

2 ASSEMBLY OF BODIES OF DIFFERENT TYPE

To use the DCA to seamlessly form and solve the equations-of-motion for the system, it is crucial that the equations of an assembled body are indistinguishable from those of a “leaf” body. Furthermore for systems with bodies of mixed definition, the equations-of-motion for any body must be in the same form as any other body and the state variables must be compatible. If these requirements can be satisfied, any method can be used to describe the deformation of the body and its internal forces in the DCA-framework. A variety of methods already meet these requirements, such as the FFR formulation using modal coordinates and associated admissible shape functions presented by Mukherjee and Anderson [19] or the FFR formulation using interpolating splines presented by Khan et al. [10].

2.1 Rigid Bodies and Flexible Bodies Limited to Small Deformation

Using an Floating Frame of Reference (FFR) approach where the deformation of Body\( ^k \) is modeled with modal coordinates (\( q \)) and admissible shape functions (\( \Phi \)), Mukherjee and Anderson [19] express the equations-of-motion as

\[
\begin{bmatrix}
\Gamma_{RR} & \Gamma_{RF} \\
\Gamma_{FR} & \Gamma_{FF}
\end{bmatrix}
\begin{bmatrix}
A_1 \\
\dot{\dot{q}}
\end{bmatrix}
= \begin{bmatrix}
\gamma_0 \\
\gamma_0
\end{bmatrix} F_{1c} + \begin{bmatrix}
\gamma_0 \\
\gamma_0
\end{bmatrix} F_{2c} + \begin{bmatrix}
\beta_0 \\
\beta_0
\end{bmatrix}.
\]

(13)
The \( \Gamma \) terms are the mass matrix sub-matrices after partitioning, \( \Gamma_{rr} \) is the typical mass-matrix associated with a rigid-body, \( \Gamma_{FF} \) is the mass matrix associated with the flexible-body motion, and the off-diagonal sub-matrices couple the rigid-body and flexible-body motion.

Importantly, the rigid-body acceleration can be algebraically decoupled from the modal accelerations associated with the flexible degrees-of-freedom. From the lower matrix equation the modal accelerations can be written as

\[
\ddot{\mathbf{q}}^k = (\Gamma_{FF})^{-1} \left( -\Gamma_{rk} \mathbf{A}^k + \gamma_{1i} \dot{q}_{1i}^k + \gamma_{1i} \dot{q}_{1i}^k + \beta_{1i}^k \right).
\]  

By substituting Eqn. (14) into the upper matrix equation of Eqn. (13), an expression for the rigid-body acceleration as a function of only the constraint forces (\( F_{k1}^1, F_{k2}^1 \)) can be written as

\[
\mathbf{A}^k_1 = \zeta_{11} F_{k1}^1 + \zeta_{13} F_{k2}^1 + \zeta_{13}.
\]  

The acceleration of \( H_2 \) can then be expressed by the kinematic relationship with \( H_1 \) as

\[
\mathbf{A}^k_2 = \left[ r \times \right]^{21} \mathbf{A}^k_1 + \mathbf{A}^k_{2/1} + \mathbf{P}_{\mathbf{H}_2} \ddot{q}^k,
\]  

where \([r \times]^{21}\) is the matrix that, when multiplied with \( \mathbf{A}^k_1 \), produces the appropriate \( \ddot{\alpha}^k \times \ddot{\rho}^{11} \) terms and the subscript \( H_2 \) of \( \mathbf{P}_{\mathbf{H}_2} \), indicates that the shape functions are evaluated at \( H_2 \). \( \mathbf{A}^k_{2/1} \) contains the relative acceleration terms that are velocity dependent, such as the centrifugal, Coriolis, and modal speed terms. Again the acceleration of \( H_2 \) can be algebraically decoupled from the modal accelerations after substituting Eqns. (14 & 15) into Eqn. (16) and grouping the coefficients of the constraint forces as

\[
\mathbf{A}^k_2 = \zeta_{21} F_{k1}^1 + \zeta_{23} F_{k2}^1 + \zeta_{23}.
\]  

In the resulting equations-of-motion for the handles on \( Body^k \), Eqns. (15 & 17), the translational and rotational accelerations are decoupled from the modal accelerations and are of exactly the same form as the equations-of-motion for a rigid-body, although they are fundamentally different. Due to this decoupling of modal accelerations and rigid-body accelerations, the assembly process can be performed for a flexible-body and a rigid-body. This is because the generalized coordinates used to describe the accelerations of the handles are compatible. More specifically, when a rigid-body \((k)\) is connected to a flexible-body \((k+1)\) at joint \( j \) there is no problem expressing the relative motion at the joint with the kinematic constraint

\[
\mathbf{A}^{rigid}_2 = \begin{bmatrix} \ddot{\alpha}^k_1 \\ \ddot{\alpha}^k_2 \end{bmatrix} = \mathbf{A}^{flex}_1 = \begin{bmatrix} \ddot{\alpha}^{k+1}_1 \\ \ddot{\alpha}^{k+1}_2 \end{bmatrix} + \dot{P}^i u + P^i \dot{u}.
\]  

Using Eqn. (18) and that the constraint forces are equal and opposite, the equations of motions for a rigid and flexible body connected by a kinematic joint \( j \) can be reduced from

\[
\mathbf{A}^{rigid}_1 = \zeta_{11} F_{k1}^1 + \zeta_{13} F_{k2}^1 + \zeta_{13} \]  

\[
\mathbf{A}^{rigid}_2 = \zeta_{21} F_{k1}^1 + \zeta_{23} F_{k2}^1 + \zeta_{23}
\]

and

\[
\mathbf{A}^{flex}_1 = \zeta_{11} F_{k1}^1 + \zeta_{12} F_{k2}^1 + \zeta_{13} \]  

\[
\mathbf{A}^{flex}_2 = \zeta_{21} F_{k1}^1 + \zeta_{22} F_{k2}^1 + \zeta_{23}
\]

to the two equations of motion for the outboard handles

\[
\mathbf{A}^{mixed}_1 = \zeta_{11}^{k+1} F_{k1}^1 + \zeta_{12}^{k+1} F_{k2}^1 + \zeta_{13}^{k+1}
\]

\[
\mathbf{A}^{mixed}_2 = \zeta_{21}^{k+1} F_{k1}^1 + \zeta_{22}^{k+1} F_{k2}^1 + \zeta_{23}^{k+1}
\]
Identical assembly and disassembly processes to that used for rigid bodies presented by Mukherjee and Anderson [19] or Featherstone [6] can be used to assemble and disassemble the inverse inertial terms. Once the flexible-body has been disassembled from the other bodies, the modal accelerations can be determined using Eqn. (14).

Similarly, this process could be done when the deformations are modeled using interpolating splines as presented by Khan et al. [10] in place of the modal coordinates and admissible shape functions. Interpolating splines would be particularly useful when modeling molecular systems with flexible multibody systems where admissible shape functions are not easily identified.

### 2.2 Rigid Bodies and Highly Flexible Bodies Undergoing Large Deformation

This section investigates the assembly and disassembly operations for rigid-bodies and flexible-bodies undergoing large deformations. This process is examined for the two cases where the deformation of the body is modeled using ANCF and GEBF. Similar to the procedure above, the equations-of-motion for the handles of a highly-flexible body are developed so the DCA can be used to form and solve the equations-of-motion for the entire system.

#### 2.2.1 Absolute Nodal Coordinate Formulation

Due to the straightforward implementation, Absolute Nodal Coordinate Formulation (ANCF) is an attractive and popular method for modeling large translations, rotations, and deformations of flexible bodies. The equations of motion for the handles of an ANCF-body have been developed by Khan and Anderson [12] so that DCA can be used to form and solve the equations of motion for the system. Khan and Anderson [12] restricted types of bodies comprising the system to contain only ANCF-bodies.

As the name implies, the ANCF uses absolute coordinates ($\vec{r}_i$) to describe the position of the $i^{th}$ node in the body and global slopes ($\frac{\partial \vec{r}_i}{\partial x}$) to describe the curvature of the body. The generalized coordinates used to describe curvature and position are

$$ e_i = \begin{bmatrix} \frac{\partial \vec{r}_i}{\partial x} & \frac{\partial \vec{r}_i}{\partial y} & \frac{\partial \vec{r}_i}{\partial z} & r'_x & r'_y & r'_z \end{bmatrix}^T. $$

Using ANCF to model the deformation of a two-node beam element, Khan and Anderson [12] give the equations-of-motion for each node in the matrix-equation

$$ \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \begin{bmatrix} \ddot{e}_1 \\ \ddot{e}_2 \end{bmatrix} = \begin{bmatrix} \lambda_{11} \\ \lambda_{21} \end{bmatrix} F_{1c} + \begin{bmatrix} \lambda_{12} \\ \lambda_{22} \end{bmatrix} F_{2c} + \begin{bmatrix} \lambda_{13} \\ \lambda_{23} \end{bmatrix} \dot{P}_{ju} + \dot{P}_j. $$

From Eqn. (26) the equations-of-motion for a node can be written as only a function of constraint and body forces as

$$ \ddot{e}_1 = \zeta_{11} F_{1c} + \zeta_{12} F_{2c} + \zeta_{13} $$

and

$$ \ddot{e}_2 = \zeta_{21} F_{1c} + \zeta_{22} F_{2c} + \zeta_{23} $$

by algebraically manipulating both equations resulting from the shown partitioning of Eqn. (26). Now, the kinematic constraint between a rigid-body ($k$) and a one-element ANCF-body ($k + 1$) as done previously is

$$ A_2^{rigid} = \frac{\partial \vec{r}_i}{\partial \vec{a}_2} = \ddot{e}_i = \left[ \frac{\partial \vec{r}_i}{\partial \vec{a}_2} \right] + P_i u + P_i a. $$
There is no difficulty expressing the translational kinematic constraint given by the lower equation of Eqn. (29). However, the rotational kinematic constraint given by the upper equation of Eqn. (29) is a meaningless equation as is. This is because there is no readily available mapping (that the authors are aware of) between rotational acceleration ($\vec{\alpha}$) and the second time derivative of the global slopes ($\partial^n \vec{r} / \partial x$) at the connecting joint (node). This is the issue which prevents recursive assembly of an ANCF-body with a rigid-body or a flexible-body undergoing small deformations modeled with an FFR formulation.

### 2.2.2 Geometrically Exact Beam Formulation

Due to the difficulty mapping the global slopes to the rotational coordinates, assembling a rigid-body and an ANCF-body is a difficult task. The Geometrically Exact Beam Formulation (GEBF) is another popular method to describe the large translations, rotations, and deformations of a body and may be more conducive to assembly with rigid or flexible bodies. Despite the difficulties associated with the Geometrically Exact Beam Formulation (GEBF), such as the interpolation of rotational coordinates between nodes, the equations-of-motion for a GEBF-element do not immediately present a problem with the kinematic relationship of the bodies at the connecting joint (node).

This can be inferred from the equations-of-motion of a GEBF-beam section given by Bauchau [1] as

$$\dot{\vec{g}} = -\tilde{\dot{u}} \vec{n} + (\tilde{x}'_0 + \tilde{u}') \vec{n} + \vec{m}' + \vec{\tau} \tag{30}$$

and

$$\dot{\vec{n}} = \vec{n}' + \vec{f} \tag{31}$$

that give the temporal rate of change of rotational and translational momentum, respectively. In this form, it is reasonable that Eqs. (30 & 31) can be manipulated so that the Left-Hand Side (LHS) is in terms of angular and translational acceleration.

**Equations of Motion of Each Node for a Two-Node Beam Element:** Instead of attempting to manipulate Eqs. (30 & 31) into a form in terms of rotational and translational acceleration, the equations-of-motion for each node of a two-node GEBF-element are developed using a generalized version of Jourdain’s principle [8], which has been popularized by Kane [9] and Schiehlen [1]. This method requires a kinematic expression of the velocity and acceleration of any point on the beam and is developed herein.

The GEBF uses a set of relative coordinates to describe the position of any point. These coordinates are the distance along the central axis of the beam ($s_1$) that locates a rigid cross-section, and the coordinates that locate the point in the cross-section ($s_2$ and $s_3$). Using this coordinate system, any point on the beam ($\vec{r}_p$) can be located as

$$\vec{r}_p = \vec{r}_0 (s_1) + s_2 \hat{a}_2 + s_3 \hat{a}_3 \tag{32}$$

where $\vec{r}_0$ is the position of the centroid of the cross-section, the $\hat{a}_i$ vectors are the unit vectors defining reference frame ‘A’ in the cross-section of the undeformed configuration with $\hat{a}_1$ orthogonal to the cross-section. The deformation of the beam can be described relative to the reference configuration by translation of the centroid of the cross-section, $u(\alpha_i)$, and the rotation of the cross-section. Expressing these vectors in the Newtonian (denoted by the subscript ‘N’) reference frame results in the position of any point on the deformed beam as

$$\{\vec{r}_p\}_N = \{\vec{r}_0\}_N + \{\tilde{u}\}_N + ^N C^B (\{\vec{s}\}_B + \{\tilde{\vec{w}}\}_B) \tag{33}$$

where

$$\{\vec{s}\}_B = 0 \hat{b}_1 + s_2 \hat{b}_2 + s_3 \hat{b}_3 \tag{34}$$
locates the position of the point in the cross-section after deformation, \( \vec{w} \) is the contribution from the warp displacement field, and \( N^b \) is the direction-cosine matrix the converts the ‘\( B \)’ basis to the ‘\( N \)’ basis. Vectors \( \vec{s} \) and \( \vec{w} \) are expressed in the basis ‘\( B \)’ which is the reference frame fixed in the deformed configuration described by unit vectors \( \hat{b}_i \) as denoted by the subscript ‘\( B \)’.

The velocity of any point (\( \vec{v}_p \)) can then be expressed in the Newtonian basis as

\[
\{ \vec{v}_p \}_N = \{ \dot{u} \}_N + N^b \{ \vec{s} \}_B
\]

if the contribution of the warp is neglected. The velocity of the point can be expressed as a function of the angular velocity of the cross-section by pre-multiplying the temporal derivative of the direction cosine matrix (\( N \dot{C}^b \)) in Eqn.(38) by the identity \( I = N^b (N^b)^T \) yielding

\[
\{ \vec{v}_p \}_N = \{ \dot{u} \}_N + I N \dot{C}^b \{ \vec{s} \}_B
\]

(36)

\[
\{ \vec{v}_p \}_N = \{ \dot{u} \}_N + N^b \left[ N \dot{\omega}^b \right]_B \{ \vec{s} \}_B
\]

(37)

In Eqn. (38) \( \left[ N \dot{\omega}^b \right]_B \) is the skew-symmetric matrix associated with the vector \( N \dot{\omega}^b \) expressed in the ‘\( B \)’ basis. Taking the time derivative of Eqn. (38) gives the expression for the acceleration of any point (\( \vec{a}_p \)) on the body as

\[
\{ \vec{a}_p \}_N = \{ \ddot{u} \}_N + N^b \left[ N \dot{\omega}^b \right]_B \left[ N \dot{\omega}^b \right]_B \{ \vec{s} \}_B + N^b \left[ N \dot{\alpha}^b \right]_B \{ \vec{s} \}_B
\]

(39)

where, similarly, \( \left[ N \dot{\alpha}^b \right]_B \) is the skew-symmetric matrix associated with the vector \( N \dot{\alpha}^b \) expressed in the ‘\( B \)’ basis.

The kinematics of all points on the GEBF-element can then be interpolated from the kinematics of the generic point shown in Eqn. (38 & 39), by multiplying with the shape function matrix \( H(x) \) resulting in

\[
\vec{a} = H(x) \{ \vec{a}_p \}_N
\]

(40)

and

\[
\vec{v} = H(x) \{ \vec{v}_p \}_N.
\]

(41)

For a two-node element the shape functions and associated matrix are

\[
h_1(x) = \frac{1}{2} (1 - x),
\]

(42)

\[
h_2(x) = \frac{1}{2} (1 + x),
\]

(43)

and

\[
H(x) = \left[ h_1(x) I, \ h_2(x) I \right]
\]

(44)

where again, \( I \) is the identity matrix. The resulting equations-of-motion for GEBF-element are

\[
\left( \rho A \right) \int_0^l \left( \vec{v}_i \cdot \vec{a} \right) dx = \sum_i \vec{v}_i \cdot \left\{ \vec{f}_{ap} + \vec{t}_{ap} + \vec{f}_{ip} + \vec{t}_{ip} + \vec{f}_{ip} + \vec{t}_{ip} \right\} + Q_e.
\]

(45)

In Eqn. (45) the \( v_i \) terms are the \( i^{th} \) partial velocities defined by taking the partial derivative of the velocity of a generic point on the GEBF-element with respect to the \( i^{th} \) generalized speed (\( \dot{e}_i \)), which is

\[
\vec{v}_i = \frac{\partial \vec{v}}{\partial \dot{e}_i}.
\]

(46)
The number of generalized coordinates for the element is \( \nu \), and \( \rho, A, l \) are the density, area, and length of the element, respectively. The vector \( Q_e \) contains the internal forces and moments of the beam element. For a GEBF-element the generalized coordinates are the translation of each node and the rotation of the cross-section at each node. For a two-dimensional problem, the generalized coordinates of the element are

\[
e = \begin{bmatrix} \theta_1 & u_{1x} & u_{1y} & \theta_2 & u_{2x} & u_{2y} \end{bmatrix}^T.
\]  

(47)

Performing the mathematical operations and collecting the coefficients of \( \ddot{q}_i \), Eqn. (45) be expressed in partitioned matrix form as

\[
\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \begin{bmatrix} \ddot{e}_1 \\ \ddot{e}_2 \end{bmatrix} = \begin{bmatrix} \gamma_{11} \\ \gamma_{12} \end{bmatrix} F_{1e} + \begin{bmatrix} \gamma_{21} \\ \gamma_{22} \end{bmatrix} F_{2e} + \begin{bmatrix} \gamma_{31} \\ \gamma_{32} \end{bmatrix}.
\]  

(48)

Equation (48) can be algebraically decomposed into an equation-of-motion for each boundary node in the element in terms of only the constraint forces at the boundaries as

\[
\ddot{e}_1 = \zeta_{11} F_{1e} + \zeta_{12} F_{2e} + \zeta_{13},
\]  

(49)

and

\[
\ddot{e}_2 = \zeta_{21} F_{1e} + \zeta_{22} F_{2e} + \zeta_{23}.
\]  

(50)

Again, Eqn. (49) and Eqn. (50) are in a form which facilitates use with the DCA and the inverse inertia terms (\( \zeta_{ij} \)) are

\[
\zeta_{11} = (M_{11} - M_{12} M_{22}^{-1} M_{21})^{-1}(\gamma_{11} - M_{12} M_{22}^{-1} \gamma_{21}),
\]  

(51)

\[
\zeta_{12} = (M_{11} - M_{12} M_{22}^{-1} M_{21})^{-1}(\gamma_{12} - M_{12} M_{22}^{-1} \gamma_{22}),
\]  

(52)

\[
\zeta_{13} = (M_{11} - M_{12} M_{22}^{-1} M_{21})^{-1}(\gamma_{13} - M_{12} M_{22}^{-1} \gamma_{31}),
\]  

(53)

\[
\zeta_{21} = (M_{22} - M_{21} M_{11}^{-1} M_{12})^{-1}(\gamma_{21} - M_{21} M_{11}^{-1} \gamma_{11}),
\]  

(54)

\[
\zeta_{22} = (M_{22} - M_{21} M_{11}^{-1} M_{12})^{-1}(\gamma_{22} - M_{21} M_{11}^{-1} \gamma_{12}),
\]  

(55)

and

\[
\zeta_{23} = (M_{22} - M_{21} M_{11}^{-1} M_{12})^{-1}(\gamma_{23} - M_{21} M_{11}^{-1} \gamma_{13}).
\]  

(56)

Due to the choice of coordinates used in Eqn. (47), the kinematic relationship of a rigid-body \( (k) \) and a GEBF-body or element \( (k+1) \) connected at a kinematic joint is expressed as

\[
A_{k+1}^{rigid} = \begin{bmatrix} \tilde{A}^k \\ \tilde{a}^k \end{bmatrix} = \begin{bmatrix} \tilde{A}^{k+1} \\ \tilde{a}^{k+1} \end{bmatrix} + \tilde{P}^j \dot{u} + P^j \ddot{u},
\]  

(57)

and the coordinates used for each body are compatible. This allows the assembly and disassembly operations to be used as if all bodies are rigid bodies. Once the flexible-body has been disassembled from other bodies the deformations of the body can be determined. For a GEBF-body this includes the troublesome task of interpolating the rotations between nodes. There have been a variety of approaches introduced to deal with this specific issue and are presented and improved upon by Bauchau [2].

3 CONCLUSIONS

With the goal to increase the capabilities of adaptive simulations beyond only changes in the number and location of degrees-of-freedom to include changes in body definition, two methods of modeling bodies undergoing large-deformations have been examined in the DCA framework. Due to the global slopes used in the ANCF method, the DCA can not be used to form and solve the
equations-of-motion for systems with bodies of mixed-definition unless a mapping between global slopes and rotational coordinates is produced.

The GEBF however, can be used without any modifications despite the difficulties typically attributed to this method. The problem of aggregating rigid and or flexible-bodies undergoing small deformations into bodies undergoing large deformations becomes an implementation task. Additionally, a metric to guide these transitions must be produced. Any body or group of bodies can be transformed from rigid to either a small-deformation flexible-body or large-deformation flexible-body or any combination thereof. Once an appropriate metric has directed the transition, the inverse inertia terms $\zeta_{ij}$ can be determined from the physical properties of the constituent bodies.

REFERENCES


Eliminating interface degrees-of-freedom in substructuring

Omid Kazemi, Parviz E. Nikravesh

Aerospace and Mechanical Engineering Department
University of Arizona
Tucson, AZ 85719, U.S.A
[kazemi, pen]@email.arizona.edu

ABSTRACT
In standard substructuring methods, which are used to reduce the order of large finite element models, a structure is partitioned into several substructures that share certain nodes (interface nodes) at their boundaries with each other. When a substructuring technique is applied to a structure, the interface nodes or a transformation of them remain in the reduced model. Keeping the interface nodes, if they are relatively high in numbers, in the condensed model reduces its computational efficiency. The computational efficiency of the condensed model could be increased if those nodes are eliminated from the model. In this paper a different method of substructuring is presented which enables the elimination of the interface DoF. The new method incorporates a different strategy in partitioning the structure into its substructures, allowing this elimination. This provides a great tool in constructing condensed finite element structures with much less total number of DoF compared to cases where a standard substructuring is used. In this paper the new method is explained for structures with two components. As an application this method is applied to a FE tire model and its results are compared to those of standard substructuring.

Keywords: Substructuring, Interface DoF, Finite element, Reduced tire model.

1 INTRODUCTION
Substructuring methods have been used in finite element (FE) analysis of large or complex structures. In these methods the FE model of the entire structure is partitioned into several smaller or less complex substructures. The FE model of each substructure is reduced individually via a component mode reduction technique and then assembled together with the remaining reduced substructures to obtain a reduced model for the entire system. Different substructuring methods in the past decades have been developed with the purpose of reducing the order of large structures [1-3]. These methods differ in the way substructures are reduced and/or in the way the reduced substructures are assembled.

Different component reduction methods are distinguished by the types of the modes that have been used in their reduction procedure. For instance the Craig-Bampton method [2], which is one of the most commonly used methods in substructuring, uses a combination of fixed-boundary normal modes and constrained (static) modes while MacNeal [3] employs free boundary normal modes and rigid body modes. The Hurty method [1] is slightly different than the Craig-Bampton method and it gives the same numerical results [4]. For a complete reference on different types of component modes, as well as component reduction techniques, the reader may choose to refer to any of these works [5-8].

Different methods of assembling the reduced substructures have been categorized based on the domains in which an assembly is performed: physical, modal, or frequency domains [5-7]. Furthermore an assembly could be performed directly on the displacement of the interface DoF or on the coupling forces that keep the substructures at an interface together [6, 7].

What is common in all of these well-known substructuring methods is that the interface DoFs, or a transformation of them, would remain in the reduced model. Although in some problems it might be necessary to keep such interface DoF in the reduced model, in general the size of the
model could be reduced even further if the interface DoFs are reduced as well. Bourquin [9] suggests to use interface modes with low frequencies instead of the set of static mode shapes which are part of the modes utilized by the Craig-Bampton method. Interface modes are defined as normal mode shapes of a condensed system which is the result of the static condensation of the structure into the interface nodes. Similar approaches have been taken in references [10, 11]. Despite the works done on the truncation of the interface DoF, none, to the best knowledge of the authors, has addressed the possibility of eliminating such DoF.

This paper discusses the possibility of eliminating the interface DoF limited to cases where displacement interface assemblies are used for structures partitioned into two substructures. It is shown that under certain conditions it is possible to eliminate the interface DoF from the reduced model. The present authors in another presentation [12] have shown the applicability of this method on a simple structure. In this paper the new method is applied to a FE tire model to isolate its contact patch from the rest of the tire. First in this paper a standard substructuring based on Craig-Bampton component model reduction for a two-component structure is reviewed. Then a new way of partitioning a structure is introduced which could allow the interface nodes to be eliminated completely from the reduced model. A FE tire model is used to demonstrate the applicability of the method on complex structures and the results are discussed.

2 A STANDARD SUBSTRUCTURING

To concentrate on the concepts, it is assumed that a structure undergoes small deformation and rotation, and it is not in contact with any other structure. We also limit our study to substructuring of a structure which is composed of two substructures as shown in Figure 1. It is assumed that a finite element analysis of each substructure has led to the following mass and stiffness matrices:

$$\begin{align*}
M_i, K_i, & \quad i = 1,2 \\
\end{align*}$$

(1)

Figure 1: Conventional way of partitioning.

For simplicity of representing the equations, damping of the substructures are ignored. Therefore, the equations of motion for each substructure are written as:

$$\begin{align*}
M_i \ddot{\delta}_i + K_i \delta_i = f_i, & \quad i = 1,2 \\
\end{align*}$$

(2)

Each substructure can be reduced individually using any of the known component model reduction techniques. Here the Craig-Bampton method with fixed-interface is used. The first step is to partition the nodes of each substructure into nodes at the interface, as the interface nodes, and the rest as the free nodes. Doing so the equations of motion of the substructures become:

$$\begin{align*}
\begin{bmatrix}
M_{ii}^{ff} & M_{ii}^{if} \\
M_{if}^{fi} & M_{ii}^{ff}
\end{bmatrix}
\begin{bmatrix}
\ddot{\delta}_i^{ff} \\
\ddot{\delta}_i^{if}
\end{bmatrix} +
\begin{bmatrix}
K_{ii}^{ff} & K_{ii}^{if} \\
K_{if}^{fi} & K_{ii}^{ff}
\end{bmatrix}
\begin{bmatrix}
\delta_i^{ff} \\
\delta_i^{if}
\end{bmatrix} = 
\begin{bmatrix}
f_i^{ff} \\
 f_i^{if}
\end{bmatrix} + 
\begin{bmatrix}
f_i^{fi} + f_i^{fin}
\end{bmatrix} \\
\quad i = 1,2
\end{align*}$$

(3)
where $f^{bb}_i$ is the coupling force at the interface acting on substructure $i$. The next step is to obtain two sets of mode shapes that Craig-Bampton utilizes: the interface-fixed normal modes and the static modes. The normal modes, $\Psi_i$, are obtained by fixing the interface nodes of each substructure and solving the corresponding generalized eigen-value problem:

$$(-\omega_i^2 M_i^{ff} + K_i^{ff}) \Psi_i = 0$$  (4)

The static mode shapes can be obtained from the stiffness matrix as

$$\Psi^{iss} = -K_i^{ff}^{-1} K_i^{fb}$$  (5)

The Craig-Bampton method suggests the following transformation of the DoF of the substructures:

$$\begin{bmatrix} \delta^b_i \\ \delta^b_i \end{bmatrix} = \begin{bmatrix} \Psi^T_i & \Psi^{iss}_i \\ 0 & I \end{bmatrix} \begin{bmatrix} z_i \\ \delta^b_i \end{bmatrix}, \quad i = 1, 2$$  (6)

Using this equation, the equations of motion of each substructure after simplification and assuming no external forces are reduced to

$$\begin{bmatrix} M^{zz}_i & M_i^{zb} \\ M_i^{bz} & M_i^{bb} \end{bmatrix} \begin{bmatrix} \dot{z}_i \\ \dot{\delta}^b_i \end{bmatrix} + \begin{bmatrix} K_i^{zz} & K_i^{zb} \\ K_i^{bz} & K_i^{bb} \end{bmatrix} \begin{bmatrix} z_i \\ \delta^b_i \end{bmatrix} = \begin{bmatrix} 0 \\ \delta^{b\text{in}} \end{bmatrix}$$  (7)

or in a simpler form as

$$\begin{bmatrix} M_i^{zz} & M_i^{zb} \\ M_i^{bz} & M_i^{bb} \end{bmatrix} \begin{bmatrix} \dot{z}_i \\ \dot{\delta}^b_i \end{bmatrix} + \begin{bmatrix} K_i^{zz} & K_i^{zb} \\ K_i^{bz} & K_i^{bb} \end{bmatrix} \begin{bmatrix} z_i \\ \delta^b_i \end{bmatrix} = \begin{bmatrix} 0 \\ \delta^{b\text{in}} \end{bmatrix}$$  (8)

After truncating each substructure, the reduced substructures need to be assembled. There are different ways of enforcing the coupling between the substructures at their interface. The most convenient way to enforce the coupling between substructures is to enforce the assembly at the displacement of the interface; i.e. nodes of substructure 1 at the interface must have the same displacements as the corresponding nodes of substructure 2 at the interface. That means $\delta^{b_1} = \delta^{b_2}$ and $f^{b\text{in}}_1 = -f^{b\text{in}}_2$. Enforcing these coupling conditions, the equations of motion of the reduced structure are obtained as

$$\begin{bmatrix} \dot{z}^{1z} \\ \dot{z}^{2z} \\ \delta^b_1 \\ \delta^b_2 \end{bmatrix} + \begin{bmatrix} M_1^{zz} & M_1^{zb} \\ 0 & M_2^{zz} \\ M_1^{bz} & M_2^{bz} + M_2^{bb} \end{bmatrix} \begin{bmatrix} z^1 \\ z^2 \\ \delta^b_1 \\ \delta^b_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \delta^{b\text{in}} \end{bmatrix}$$  (9)

where

$$M = \begin{bmatrix} M_1^{zz} & 0 & M_1^{zb} \\ 0 & M_2^{zz} & M_2^{zb} \\ M_1^{bz} & M_2^{bz} & M_1^{bb} + M_2^{bb} \end{bmatrix} \quad K = \begin{bmatrix} K_1^{zz} & 0 & K_1^{zb} \\ 0 & K_2^{zz} & K_2^{zb} \\ K_1^{bz} & K_2^{bz} & K_1^{bb} + K_2^{bb} \end{bmatrix}$$  (10)

Another way of obtaining a reduced model is to assemble the two substructures before truncation. In other words one can start with the unreduced equations of motion of the entire structure as
Then we can apply the following transformation based on the Craig-Bampton model reduction

\[
\begin{bmatrix}
M_1^{ff} & 0 & M_1^{bb} \\
0 & M_2^{ff} & M_2^{bb} \\
M_1^{bf} & M_2^{bf} & M_1^{bb} + M_2^{bb}
\end{bmatrix}
\begin{bmatrix}
\delta_i^f \\
\delta_i^s \\
\delta_i^b
\end{bmatrix} +
\begin{bmatrix}
K_1^{ff} & 0 & K_1^{bb} \\
0 & K_2^{ff} & K_2^{bb} \\
K_1^{bf} & K_2^{bf} & K_1^{bb} + K_2^{bb}
\end{bmatrix}
\begin{bmatrix}
\delta_i^f \\
\delta_i^s \\
\delta_i^b
\end{bmatrix} =
\begin{bmatrix}
0 \\
0 \\
0
\end{bmatrix}
\tag{11}
\]

Using this transformation it can be shown that the same equations of motion described by Equation (9) can be obtained. This approach will be used in the next section as a base for elimination of the interface DoF.

3 ELIMINATING INTERFACE DOF

To eliminate the interface DoF which are left in the reduced model when standard substructuring methods are used, different partitioning of the structure is used. Here the partitioning is performed in such a way that the two substructures overlap beyond a surface. As seen in Figure 2, the intersection of the two substructures is no longer a border plane but a sub-volume of the main structure. Each node of this intersection is either a interface node of substructure 1 or an interface node of substructure 2. The two substructures do not have any other interface nodes that are not part of the intersection. This form of partitioning dictates that the interface nodes of each substructure to be a subset of the unbounded nodes of the other substructure. The nodes of the flexible-body, as shown in Figure 3, can be partitioned into four sets that their mutual intersections are empty:

- free nodes of substructure 1 (f1-nodes); all the unbounded nodes of substructure 1 that are not interface nodes of substructure 2,
- free nodes of substructure 2 (f2-nodes); all the unbounded nodes of substructure 2 that are not interface nodes of substructure 1,
- the interface nodes of substructure 1 (b1-nodes), and
- the interface nodes of substructure 2 (b2-nodes).

**Figure 2:** New way of partitioning.

\[
\begin{bmatrix}
\Psi^i & 0 & \Psi^{is} \\
0 & \Psi^s & \Psi^{ss} \\
0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
z^i \\
z^s \\
\delta^b
\end{bmatrix}
\tag{12}
\]
Figure 3: Partitioning of the nodes in the new method of substructuring.

Using the Craig-Bampton component mode synthesis, the following transformations can be written for the unbounded nodes of the two substructures:

\[ \delta'' = \Psi^i z^i + \Psi^{1s} \delta^b_i, \quad \delta'' = \Psi^2 z^2 + \Psi^{2s} \delta^b_i \]  

(3.13)

where, \( \Psi^i \) and \( \Psi'' \) are the normal and the static mode shapes of substructure \( i \) \((i = 1, 2)\). For each substructure the normal modes are obtained with their interface nodes constrained. Expanding the above equations we obtain

\[ \begin{bmatrix} \delta'_1 \\ \delta'_2 \\ \delta''_1 \\ \delta''_2 \end{bmatrix} = \begin{bmatrix} \Psi^i_f & 0 & \Psi^{1s}_f & 0 \\ 0 & \Psi^2_f & 0 & \Psi^{2s}_f \\ 0 & \Psi^i_b & 0 & \Psi^{1s}_b \\ \Psi^i_b & 0 & \Psi^{1s}_b & 0 \end{bmatrix} \begin{bmatrix} z^i \\ z^2 \\ \delta''_1 \\ \delta''_2 \end{bmatrix} \]  

(3.14)

These transformations can be combined to obtain the following:

\[ \begin{bmatrix} \delta'_1 \\ \delta'_2 \\ \delta''_1 \\ \delta''_2 \end{bmatrix} = \begin{bmatrix} \Psi^i_f & 0 & \Psi^{1s}_f & 0 \\ 0 & \Psi^2_f & 0 & \Psi^{2s}_f \\ 0 & \Psi^i_b & 0 & \Psi^{1s}_b \\ \Psi^i_b & 0 & \Psi^{1s}_b & 0 \end{bmatrix} \begin{bmatrix} z^i \\ z^2 \\ \delta''_1 \\ \delta''_2 \end{bmatrix} \]  

(3.15)

Bringing \( \delta''_1 \) and \( \delta''_2 \) from the right side of the equation to the left side yields:

\[ \begin{bmatrix} 1 & 0 & -\Psi^{1s}_f & 0 \\ 0 & 1 & 0 & -\Psi^{2s}_f \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\Psi^{1s}_b & 1 \end{bmatrix} \begin{bmatrix} \delta'_1 \\ \delta'_2 \\ \delta''_1 \\ \delta''_2 \end{bmatrix} = \begin{bmatrix} \Psi^i_f & 0 \\ 0 & \Psi^2_f \\ \Psi^i_b & 0 \\ 0 & \Psi^2_b \end{bmatrix} \begin{bmatrix} z^i \\ z^2 \\ \delta''_1 \\ \delta''_2 \end{bmatrix} \]  

(3.16)

Let us call the matrix on the left-hand-side of this equation \( T \). This equation can be used to eliminate the interface nodes in the reduced model as long as the inverse of \( T \) exist; i.e.,

\[ \begin{bmatrix} \delta'_1 \\ \delta'_2 \\ \delta''_1 \\ \delta''_2 \end{bmatrix} = T^{-1} \begin{bmatrix} \Psi^i_f & 0 \\ 0 & \Psi^2_f \\ \Psi^i_b & 0 \\ 0 & \Psi^2_b \end{bmatrix} \begin{bmatrix} z^i \\ z^2 \end{bmatrix} \]  

(3.17)

This equation transforms the entire structure to a purely modal space. It should be noted that due to specific structure of matrix \( T \), its inverse can be obtained in an efficient way. It can be shown that the inverse of \( T \) exists if the following matrix is non-singular:
By direct calculation one can obtain the closed form of the inverse of $T$ as:

$$T^{-1} = \begin{bmatrix} 1 & 0 & \Psi_s^1 P^{-1} & \Psi_s^b P^{-1} \Psi_s^b \\ 0 & 1 & \Psi_s^2 P^{-1} & \Psi_s^b P^{-1} \Psi_s^b \\ 0 & 0 & P^{-1} & P^{-1} \Psi_s^b \\ 0 & 0 & \Psi_s^b P^{-1} & I + \Psi_s^b P^{-1} \Psi_s^b \end{bmatrix}. \tag{3.19}$$

We can substitute Eq. (3.19) into Eq. (3.17) to get:

$$\begin{align*}
\begin{bmatrix} \delta_1^f \\ \delta_2^f \\ \delta_1^b \\ \delta_2^b \end{bmatrix} &= \begin{bmatrix} \Psi_s^f + \Psi_s^f P^{-1} \Psi_s^b \Psi_s^b \\ \Psi_s^2 + \Psi_s^2 P^{-1} \Psi_s^b \Psi_s^b \\ P^{-1} \Psi_s^b \\ \Psi_s^b \end{bmatrix} \begin{bmatrix} \zeta_1^f \\ \zeta_1^b \end{bmatrix}, \text{or } \begin{bmatrix} z_1 \\ z_2 \end{bmatrix},
\end{align*} \tag{3.20}$$

This transformation is then used to obtain the reduced model of the structure:

$$\Phi^T M \Phi \zeta + \Phi^T K \Phi \zeta = 0 \tag{3.21}$$

4 CASE STUDY

A FE tire model is chosen for implementation of the new substructuring method. Tires, being the only components of a vehicle in contact with the ground, generate most of the external forces and moments that act on a vehicle. The modeling of tires and the accurate prediction of ground-tire interaction forces and moments have always been of interest, and yet remain a challenge in multibody simulations. Incorporating detailed FE tire models in multibody simulations of vehicles is computational expensive and infeasible. Therefore a model reduction is needed to reduce the FE tire model. The present authors in several works [13-15] have addressed the challenges of such a reduction and proposed methodologies of obtaining reduced FE tire models suitable for multibody simulations.

One way to improve a reduced FE tire model is to isolate the contact patch region from the rest of the tire. This will allow a more efficient selection of the mode shapes that can possibly capture the contact patch deformation. To isolate the contact patch region substructuring should be used. The new substructuring method and the standard substructuring method reviewed in this paper are used to partition a FE tire into two substructures: the contact patch substructure and the rest of the tire substructure. The FE tire model used in this comparison, as is seen in Figure 4, has a nonuniform mesh around its contact region and contains about 46,000 nodes. With the new and the standard substructuring methods two reduced tire models are obtained.
Figure 5 and Figure 6 show how the tire model is partitioned into two substructures when the standard and the new substructuring methods are used. Table 1 summarizes the reduced tire models that are constructed using these substructuring methods. As it can be seen from the table the interface DoF are eliminated from the reduced model based on the new substructuring method. The total DoF of the reduced model based on the new substructuring method is an order of magnitude smaller than the reduced model based on the standard substructuring method. In both reduced models 100 modes with the lowest frequencies are selected from the contact patch substructure and 25 modes with the lowest frequencies are selected from the rest of the tire.

**Figure 5**: Partitioning of the nodes in the standard method of substructuring. Red nodes are the free nodes of the contact patch SS and the black nodes are free nodes of the rest of the tire SS. The blue nodes are the interface nodes shared between the two SS.

**Figure 6**: Partitioning of the nodes in the new method of substructuring. Red nodes are the f nodes of the contact patch SS and the black nodes are f nodes of the rest of the tire SS. The blue nodes are the b nodes of the contact patch SS and the green nodes are the b nodes of the rest of the tire SS.

**Table 1. Summary of the reduced FE tire models**

<table>
<thead>
<tr>
<th>SS method</th>
<th>Standard</th>
<th>New</th>
</tr>
</thead>
<tbody>
<tr>
<td>DoF of the contact patch SS</td>
<td>5589</td>
<td>5967</td>
</tr>
<tr>
<td>DoF of the rest of the tire SS</td>
<td>42135</td>
<td>43245</td>
</tr>
<tr>
<td>Interface DoF of the contact patch SS</td>
<td>1704</td>
<td>1704</td>
</tr>
<tr>
<td>Interface DoF of the rest of the tire SS</td>
<td>1704</td>
<td>1488</td>
</tr>
<tr>
<td># of modes of the contact patch SS</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td># of modes of the rest of the tire SS</td>
<td>25</td>
<td>25</td>
</tr>
<tr>
<td>Total DoF of the reduced tire model</td>
<td>1829</td>
<td>125</td>
</tr>
</tbody>
</table>
Figure 7 shows the low natural frequencies of the reduced tire models compared to the ones of the full model (untruncated). The results suggest that the low frequencies in the reduced model with the new method match well with the untruncated model. There are differences between the natural frequencies of the reduced model with the standard substructuring and the untruncated model. The main advantage of using the new method is that the order of the truncated model is reduced substantially and still acceptable results are obtained.

![Figure 7: Low natural frequencies of the reduced models compared to the untruncated model.](image)

5 CONCLUSIONS

In this paper a different way of partitioning in substructuring is which allows for the interface DoF to be eliminated from the reduced model. The methodology was discussed in detail for structures with two components. A detailed FE tire model was used to demonstrate the applicability of the method. Reduced FE tire models with the standard and the new substructuring methods were created. The results showed that by eliminating the boundary DoF one can obtain close results to the untruncated model. The main gain here is the elimination of the interface DoF which leads to a smaller reduced model.

REFERENCES


Influence of the fibre spring-damper model in a simple laboratory mechanical system on the coincidence with the experimental results

Pavel Polach*, Michal Hajžman#

*Section of Materials and Mechanical Engineering Research
Research and Testing Institute Plzeň
Tylova 1581/46, 301 00 Plzeň, Czech Republic
polach@vzuplzen.cz

#Faculty of Applied Sciences
University of West Bohemia
Univerzitní 8, 306 14 Plzeň,
Czech Republic
mhajzman@kme.zcu.cz

ABSTRACT
Experimental measurements focused on the investigation of a fibre behaviour are performed on an assembled weigh-fibre-pulley-drive mechanical system. The carbon fibre, which is driven by one drive, is led over a pulley. On its other end there is a prism-shaped steel weight, which moves in a prismatic linkage on an inclined plane. In presented case the position of the weight is symmetric with respect to the plane of a drive-pulley symmetry. Drive excitation signals can be of different shapes with the possibility of variation of a signal rate. Dynamic responses of the weight and the fibre are measured. The same system is numerically investigated by means of multibody models. The influence of the fibre stiffness and fibre damping coefficient in the computational model on the coincidence of the simulation results and the experimental measurement results is evaluated. Time histories of the weight position and of the force acting in the fibre are the measured quantities. The simulations’ aim is to create a phenomenological model of a fibre, which will be utilizable in fibre modelling in the case of more complicated mechanical or mechatronic systems.

Keywords: Fibre, Mechanical system, Dynamic response, Phenomenological model.

1 INTRODUCTION
The replacement of the chosen rigid elements of manipulators or mechanisms by fibres or cables [1] is advantageous due to the achievement of a lower moving inertia, which can lead to a higher machine speed, and lower production costs. Drawbacks of using the flexible elements like that can be associated with the fact that cables should be only in tension (e.g. [2, 3]) in the course of a motion.

Experimental measurements focused on the investigation of the fibre behaviour are performed on an assembled weigh-fibre-pulley-drive system [4, 5, 6, 7, 8, 9]. The fibre is driven with one drive, it is led over a pulley and on its other end there is a prism-shaped steel weight, which moves on an inclined plane. The position of the weight can be symmetric (see Figure 1) or asymmetric with respect to the plane of a drive-pulley symmetry. It is possible to add an extra mass to the weight. The same system is numerically investigated using multibody models created in the alaska simulation tool [10]. The influence of the model parameters on the coincidence of the results of experimental measurements and the simulations results is evaluated. The simulation aim is to create a phenomenological model of a fibre, which will be utilizable in fibre modelling in the case of more complicated mechanical or mechatronic systems.

The fibre damping coefficient, the fibre stiffness and the friction force acting between the weight and the prismatic linkage were considered to be system parameters of the phenomenological model. The parameters determined at investigating the weight-fibre system [11, 12] were applied in the fibre model of the weight-fibre-pulley-drive system. At simulating the experimental measurements for “slower” drive motion [4, 5, 6, 7, 8] the local extremes of the time histories of the weight displacement and of the force acting in the fibre are independent of the fibre stiffness and the fibre damping coefficient (considered in feasible intervals of
values). The friction force acting between the weight and the prismatic linkage, as it has been confirmed [4, 5, 6, 7, 8], is not the parameter of the phenomenological model. This quantity is dependent on angle $\alpha$ of the inclined plane. At simulating the experimental measurements for “quicker” drive motion [4, 5, 6, 7, 8] the local extremes of the monitored time histories are dependent on both the phenomenological model parameters. From the obtained results it was evident that these parameters of the fibre phenomenological model must be, in addition, considered dependent on the velocity of the weight motion. That is why the influence of considering the velocity-dependent stiffness [9] and the velocity-dependent damping “coefficient” in the fibre model on dynamic response of the system is investigated in this paper.

![Figure 1](image1.png)

**Figure 1.** Scheme and a real weight-fibre-pulley-drive mechanical system (symmetric position of the weight).

2 EXPERIMENTAL STAND

Experimental measurements focused on the investigation of the fibre behaviour are performed on an assembled weigh-fibre-pulley-drive mechanical system (see Figure 1). A carbon fibre with a silicone coating (see e.g. [13]) is driven with one drive and it is led over a pulley. The fibre length is 1.82 meters (fibre weight is 4.95 grams), the pulley diameter is 80 millimetres. The weight position can be symmetric [4, 6] (see Figure 1) or asymmetric [5, 7] with respect to the vertical plane of drive-pulley symmetry (distance of the weight from the vertical plane of drive-pulley symmetry is 280 millimetres in the case of the asymmetric weight position). At the drive the fibre is fixed on a force gauge. In the other end of the fibre there is a prism-shaped steel weight (weight 3.096 kilograms), which moves in a prismatic linkage on an inclined plane. It is possible to add an extra mass (weight 5.035 kilograms) to the weight [6, 7]. The angle of inclination of the inclined plane can be changed. In the case of the symmetric weight position the angle is $\alpha = 30$ degrees and the pulley-fibre angle is $\varphi = 150$ degrees (in the case of the asymmetric weight position the angle is $\alpha = 30.6$ degrees and the pulley-fibre angle is $\varphi = 146$ degrees). Drive exciting signals can be of a rectangular, a trapezoidal and a quasi-sinusoidal shape and there is a possibility of variation of a signal rate [14]. The amplitudes of the drive displacements are up to 90 millimetres. Time histories of weight position $u$ (in direction of the inclined plane; measured by means of a dial gauge), of drive position $x$ (in vertical direction) and of the force acting in the fibre (measured by a force gauge at drive) are recorded using a sample rate of 2 kHz.
3 POSSIBILITIES OF THE FIBRE MODELLING

The fibre (cable, wire etc.) modelling [15] should be based on considering the fibre flexibility and suitable approaches can be based on the flexible multibody dynamics (see e.g. [16, 17]). Flexible multibody dynamics is a rapidly growing branch of computational mechanics and many industrial applications can be solved using newly proposed flexible multibody dynamics approaches. Studied problems are characterized by a general large motion of interconnected rigid and flexible bodies with the possible presence of various nonlinear forces and torques. There are many approaches to the modelling of flexible bodies in the framework of multibody systems [18]. Comprehensive reviews of these approaches can be found in [16] or in [19]. Further development together with other multibody dynamics trends was introduced in [20].

The simplest way how to incorporate fibres in equations of motion of a mechanism is the force representation of a fibre (e.g. [21]). It is assumed that the mass of fibres is low to such an extent comparing to the other moving parts that the inertia of fibres is negligible with respect to the other parts. The fibre is represented by the force dependent on the fibre deformation and its stiffness and damping properties. This way of the fibre modelling is probably the most frequently used one in the cable-driven robot dynamics and control (e.g. [22, 23]). The fibre-mass system fulfils all requirements for modelling the fibre using the force representation of the fibre. A more precise approach is based on the representation of the fibre by means of a point-mass model (e.g. [24]). It has the advantage of a lumped point-mass model. The point masses can be connected by forces or constraints.

The massless fibre model is considered in this phase of investigation of the weight-fibre-pulley-drive system. The fibre model is considered to be phenomenological and it is modelled by the forces which comprise e.g. influences of fibre transversal vibration, “jumping” from pulley etc. The multibody models of the weight-fibre-pulley-drive system in the case of considering the symmetric and asymmetric position of the weight with respect to the plane of drive-pulley symmetry slightly differ [4, 5]. In the case of symmetric position the number of degrees of freedom in kinematic joints is 5 (in the case of asymmetric position the number of degrees of freedom in kinematic joints is 6). The weight (with added mass), the pulley and the drive are considered to be rigid bodies. A planar joint between the weight and the base (prismatic linkage), a revolute joint between the pulley and the base and a prismatic joint between the drive and the base (the movement of the drive is kinematically prescribed) are considered. Behaviour of this nonlinear system is investigated using the alaska simulation tool [10].

4 SIMULATION AND EXPERIMENTAL RESULTS

As it has already been stated the simulations aim was to create a phenomenological model of a fibre. When looking for compliance of the results of experimental measurement with the results of simulation influences of the fibre stiffness and the fibre damping coefficient are considered. The friction force acting between the weight and the prismatic linkage in which the weight moves was considered to be phenomenological model parameter in the first phase of investigation [11, 12].

Investigation of the (carbon) fibre properties eliminating the influence of the drive and of the pulley was an intermediate stage before the measurement on the stand [11, 12]. A phenomenological model dependent on the fibre stiffness, on the fibre damping coefficient and on the friction force acting between the weight and the prismatic linkage was the result of this investigation. When looking for the fibre model [12] that would ensure the similarity of time histories of the weight displacement and time histories of the dynamic force acting in a fibre as high as possible a fibre stiffness and a fibre damping coefficient were considered to be constant in this phase of the fibre behaviour research. The friction force course (in dependence on the weight velocity) was considered nonlinear (basis for the determination of the friction force course was especially [25] and [26]). Values of fibre stiffness and fibre damping coefficient were calculated on the basis of the values determined in [4] (see Table 1). The friction force course determined at investigating the weight-fibre mechanical system [11] with
the angle of inclination of the inclined plane 30 degrees (see Figure 2) is applied in the model of the weight-fibre-pulley-drive mechanical system [4, 5, 6, 7, 8].

Figure 2. Friction force acting between the weight and the prismatic linkage.

At simulating the experimental measurements for a “quicker” drive motion [4, 5, 6, 7, 8] the local extremes of the time histories of measured and calculated quantities were more or less different (to explain: frequencies of drive motion – i.e. frequencies of input signal – higher than 1 Hz - are designated as “quicker” drive motions, frequencies of drive motion lower than 1 Hz are designated as “slower” drive motions). From these results it was evident that the parameters of the fibre phenomenological model must be, in addition, considered dependent on the velocity of the weight motion. That is why the influence of considering the velocity-dependent stiffness and the velocity-dependent damping coefficient in the fibre model on dynamic response of the system is investigated.

Velocity-dependent stiffness \( c \) of the fibre is supposed in the form

\[
\begin{align*}
c & = \begin{cases} 
c_c, & \text{if } v \leq v_{tr} \\
c_c + (v - v_{tr}) \cdot c_2, & \text{if } v > v_{tr}
\end{cases},
\end{align*}
\]

(1)

where \( c_c \) is constant fibre stiffness (taken from [4]), \( c_2 \) is constant, \( v \) is instantaneous velocity of the weight and \( v_{tr} \) is threshold value of the velocity of the weight [9]. Optimal (constant) values of constant \( c_2 \) and threshold value of weight velocity \( v_{tr} \) are found.

Velocity-dependent damping “coefficient” \( b \) of the fibre is considered similarly as the velocity-dependent stiffness

\[
\begin{align*}
b & = \begin{cases} 
b_c, & \text{if } v \leq v_{tr} \\
b_c + (v - v_{tr}) \cdot b_2, & \text{if } v > v_{tr}
\end{cases},
\end{align*}
\]

(2)

where \( b_c \) is constant fibre damping coefficient (taken from [4]), \( b_2 \) is constant. The optimal (constant) value of constant \( b_2 \) is found.

The influence of the fibre stiffness and the fibre damping “coefficient” values on time histories of the weight displacement and also on time histories of the dynamic force acting in the fibre is evaluated partly visually and partly on the basis of the value of the correlation coefficient between the records of the experimental measurements and the simulation results. Application of the approach based on the calculation of the statistical quantities that enables to express...
directly the relation between two time series has appeared to be suitable for comparing two time series in various cases – e.g. [27, 28].

Correlation coefficient $R(p)$ [29] defined for two discrete time series $x(t)$ (the time history recorded at experimental measurement) and $x(t)(p)$ (the time history determined at simulation with the multibody model; function of investigated parameters $p$) was calculated:

$$R(p) = \frac{\sum_{i=1}^{n} (x(t) - \mu_1) \cdot [x(t)(p) - \mu_2]}{\sqrt{\sum_{i=1}^{n} (x(t) - \mu_1)^2 \cdot \sum_{i=1}^{n} [x(t)(p) - \mu_2]^2}},$$

(3)

where $\mu_1$ and $\mu_2(p)$ are mean values of the appropriate time series. The maximum value of the correlation coefficient is 1. The more the compared time series are similar to each other the more the correlation coefficient tends to 1. The advantage of the correlation coefficient is that it quantifies very well the similarity of two time series by scalar value, which is obtained using a simple calculation.

The problem can be put as the problem of the minimization of the objective function in the form

$$\psi(p) = (1 - R(p))^2.$$  

(4)

In case of the computer simulations in the alaska 2.3 simulation tool, the whole process of the optimization was limited by the impossibility of executing the analysis from the statement line and evaluating the results of numerical simulations without the necessary human intervention. The whole process could not be automated. “Manual” change in the parameters on the basis of the chosen optimization method was the only solution. Comparing to automated optimization process it is not possible to perform so many iteration cycles in a short time. But the advantage is that during the evaluation it is possible to respect criteria that do not have to be strictly mathematically formulated (the coefficient of correlation given by Equation (3) enables to imagine coincidence of (time) series, but it is not “universal”).

In Table 1 there are given the optimal values of parameters in Equations (1) and (2) of the investigated model of the weight-fibre-pulley-drive mechanical system. From Table 2 it is evident that the values of correlation coefficient $R(p)$ are “better” for the determined velocity-dependent stiffness and velocity-dependent damping “coefficient” than for the constant values of fibre model parameters. In the time histories of the dynamic force acting in fibre the correlation coefficient $R(p)$ improvement is not evident very much. At time histories of dynamic force acting in the fibre during determining the optimum values of velocity-dependent parameters attention was paid especially to keeping the character of their course and achieving the best possible agreement of extreme values of the measured and the calculated dynamic forces.

**Table 1.** Values of stiffness and damping coefficients of the fibre model.

<table>
<thead>
<tr>
<th></th>
<th>Threshold value of the velocity $[m/s^2]$</th>
<th>Stiffness $[N/m]$</th>
<th>Damping coefficients $[N\cdot s/m]$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$v_{tr}$</td>
<td>$c_e$</td>
<td>$c_2$</td>
</tr>
<tr>
<td>Constant coefficients</td>
<td>-</td>
<td>$34 \cdot 10^3$</td>
<td>-</td>
</tr>
<tr>
<td>Force-dependent “coefficients”</td>
<td>0.4</td>
<td>$34 \cdot 10^3$</td>
<td>850</td>
</tr>
</tbody>
</table>

At “slower” drive motions the time histories of the weight displacement recorded at the experimental measurements and computed at the computer simulations are approximately identical (see [4, 5, 6, 7, 8]) and hence it is not desirable to change values of parameters of the fibre phenomenological model. It is evident that it was the reason why the threshold value of...
velocity \( v_{tr} \) of the weight was determined in such a way that the maximum velocity of the weight during “slower” drive motions might be lower than \( v_{tr} \).

Table 2. Values of correlation coefficient \( R(p) \) [-].

<table>
<thead>
<tr>
<th>Tested situation</th>
<th>Comparison of the time histories of the weight displacement</th>
<th>Comparison of the time histories of the dynamic force acting in fibre</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Constant stiffness and damping coefficient</td>
<td>Velocity-dependent stiffness and damping “coefficient”</td>
</tr>
<tr>
<td>2</td>
<td>0.9929</td>
<td>0.9934</td>
</tr>
<tr>
<td>3c</td>
<td>0.7552</td>
<td>0.9464</td>
</tr>
<tr>
<td>7a</td>
<td>0.9999</td>
<td>0.9999</td>
</tr>
<tr>
<td>10</td>
<td>0.9834</td>
<td>0.9836</td>
</tr>
<tr>
<td>11</td>
<td>0.2187</td>
<td>0.6534</td>
</tr>
</tbody>
</table>

Figure 3. Time histories at a “quicker” tested situation (symmetric position of the weight, the weight without added mass), a) weight displacement, b) dynamic force acting in a fibre.

Figure 4. Time histories at a “quicker” tested situation (symmetric position of the weight, the weight without added mass), a) weight displacement, b) dynamic force acting in a fibre.
Figure 5. Time histories at a “quicker” tested situation (symmetric position of the weight, the weight with added mass), a) weight displacement, b) dynamic force acting in a fibre.

Figure 6. Time histories at a “quicker” tested situation (symmetric position of the weight, the weight with added mass), a) weight displacement, b) dynamic force acting in a fibre.

Figure 7. Time histories at a “slower” tested situation (symmetric position of the weight, the weight without added mass), a) weight displacement, b) dynamic force acting in a fibre.
Results of experimental measurements and simulations of five selected tested situations at weight symmetric position are presented in this paper (see Figures 3 to 7). Four tested situations are at a “quicker” drive motion (see time histories of drive motion in Figure 3a, Figure 4a, Figure 5a, and Figure 6a) and one situation is at a “slower” drive motion (see time history of drive motion in Figure 7a).

General pieces of knowledge obtained at investigating the weight-fibre-pulley-drive system, independently of the combination of the position of the weight with respect to the plane of the drive-pulley symmetry (symmetric or asymmetric) and of the mass of the weight (without or with added mass), are similar.

As it has already been stated in [6] and [8], the highest frequency of drive motion (i.e. the highest frequency of input signal) at investigation of the weight-fibre-pulley-drive system is 2 Hz (see Figure 3a, Figure 4a, Figure 5a and Figure 6a). This frequency of drive motion is much lower than natural frequencies of the computer model of linearized system in an equilibrium position. Natural frequency corresponding to the weight vibrations of the system with weight without added mass is 25 Hz and natural frequency of the system with weight with added mass is 15.25 Hz. It means that in case of weight vibration at “quicker” tested situations the excitation of resonant vibrations is not concerned, but vibrations that are given by strongly nonlinear behaviour of a fibre (as it has been already stated, fibres are able to transfer only tensile force, in “compression” they are not able to transfer any force), which can even have the character of chaos, are involved.

Time histories of the weight displacement recorded at the experimental measurements and computed at the computer simulations at “slower” tested situations are approximately identical and it was not necessary to consider the values of the parameters of the fibre phenomenological model velocity-dependent (see Figure 7a).

At the “quicker” tested situations the measured and the computed time histories of the weight displacement are of the same character (see Figure 3a, Figure 4a, Figure 5a and Figure 6a). It is evident that owing to considering the velocity-dependent stiffness and the velocity-dependent damping “coefficient” in the fibre model the agreement of the measured and the computed time histories and the local extremes of the weight displacement has improved (see Figure 3a, Figure 4a, Figure 5a and Figure 6a).

As it has already been stated in [4, 5, 6, 8], at all the simulations when changing the computational model the time histories of a dynamic force acting in the fibre are different (more or less) but their character remains the same. From Figure 3b, Figure 4b, Figure 5b, Figure 6b and Figure 7b it is evident that time histories of dynamic force acting in the fibre are not suitable for determining the parameters of the fibre phenomenological model. It follows from the fact, that the phenomenological model of a fibre is to cover, as it has been stated, e.g. influences of the fibre transversal vibration, “jumping” from the pulley etc. As it does not include those phenomena physically (but by the change in the already introduced model parameters), it is evident, that the introduced time histories of the dynamic force acting in the fibre cannot be expected to be of the same course.

For searching for the parameters of the fibre phenomenological model it is necessary to use the results of experimental measurements with the “quicker” drive motion. The possibility of performing experimental measurements with other time histories of drive motion or with a different geometrical arrangement of the experimental stand will be analysed.

5 CONCLUSIONS

The approach to the fibre modelling based on the force representations was utilised for the investigation of the weight motion in the weigh-fibre-pulley-drive mechanical system. The simulation aim is to create a phenomenological model of the fibre, which will be utilizable in fibre modelling in the case of more complicated mechanical or mechatronic systems. The
created phenomenological model is assumed to be dependent on the velocity-dependent fibre stiffness and on the velocity-dependent fibre damping “coefficient”.

Development of the fibre phenomenological model continues. It can be supposed that in a more sophisticated phenomenological model of the fibre more complicated dependencies of the fibre stiffness and of the fibre damping “coefficient” on the weight velocity will be considered.

In addition it must be stated that the model of the fibre-pulley contact appears to be problematic in the computational model.

The paper has originated in the framework of institutional support for the long-time conception development of the research institution provided by the Ministry of Industry and Trade of the Czech Republic to Research and Testing Institute Plzeň and solving No. 15-20134S project of the Czech Science Foundation entitled “Multi-level Light Mechanisms with Active Structures”.

REFERENCES


Interpolation and Truncation Model Reduction Techniques in Coupled Elastic Multibody Systems

Philip Holzwarth, Peter Eberhard

Institute of Engineering and Computational Mechanics
University of Stuttgart
Pfaffenwaldring 9, 70569 Stuttgart, Germany
[philip.holzwarth,peter.eberhard]@itm.uni-stuttgart.de

ABSTRACT

Many technical systems are designed in a modular fashion. The division of tasks leads to systems with components that can be separated and reused. Elastic multibody systems are very well suited to describe modular systems of this kind. However, the dynamic behavior of each single elastic body is subject to its surrounding environment and boundary conditions. The goal is, therefore, to find a representation of the elasticity of each body that can be used in different settings. The elasticity model is derived with the finite element method. In order to keep the computational burden at an acceptable level, model order reduction is performed. Many of the reduction methods used in this context can be classified in two categories, truncation methods and interpolation methods. It will be shown that for connected systems, the moment matching conditions which are introduced with interpolation methods for single components also hold for an assembled system. At the same time, error bounds that are guaranteed for balanced truncation methods do not hold anymore after the introduction of connections to the environment. The combination of interpolatory and truncation-based model order reduction, which is e.g. used in the well known Craig-Bampton method, can be generalized and the use of Gramian-matrix based truncation methods is suited to improve the accuracy of multibody simulations.

Keywords: Model Order Reduction, Elastic Multibody System, Component Mode Synthesis, Substructuring.

1 INTRODUCTION

For many applications, substructuring of dynamical systems is useful. The division of an overall assembly into single components can drastically decrease the numerical effort for the solution or enable parallelized algorithms. For the verification of a numerical model, measurements of dynamical systems are often only possible if they are decomposed into single entities. During the design process of a technical product often small modifications are carried out on component level, for example in an optimization process. Databases consisting of single substructures can be created and reused for many different settings.

Using the floating frame of reference approach in elastic multibody systems (EMBS), the kinematics of each body are split into a rigid body motion and a deformation. This allows a nonlinear description of the rigid body motion and naturally gives rise to a substructured system. The elasticity description is often based on a fine spatial discretization. This leads to models of high dimension and model order reduction (MOR) is performed on each elastic body to make time integrations feasible. The key step is finding a reduced order model that approximates the important dynamical properties. However, for changing environmental settings these properties may change. Typically bushings and constraints connecting the body to other bodies or the inertial system are introduced.

In this contribution it is shown that for interpolation-based reduction methods like Krylov-subspace methods or Guyan condensation, the interpolation of the transfer function maintains valid independent of the environmental settings. Truncation-based methods like modal or balanced truncation in contrast lose key aspects, e.g. eigenmodes respectively the balanced realization are not maintained for coupled systems if derived for decoupled subsystems.
2 ELASTIC MULTIBODY SYSTEMS

Elastic multibody dynamics is based on the kinematic and kinetic description of a system of rigid and elastic bodies. Using the floating frame of reference approach, the motion of an arbitrary point of each body is split into a rigid body motion, nonlinearly described by the movement of the reference frame, and a superposed elastic deformation, which is described linearly and relative to the reference frame, see [1] for details. Using the principles of mechanics, e.g. the principle of virtual work, the equation of motion for a single body reads

\[
\begin{bmatrix}
    mI & m\dot{e}^T(q) & C_i^T(q) \\
    m\dot{e}^T(q) & J(q) & C_i^T(q) \\
    C_i(q) & C_i(q) & M
\end{bmatrix}
\begin{bmatrix}
    \nu_{IR} \\
    \dot{\omega}_{IR} \\
    \dot{q}
\end{bmatrix}
= -\begin{bmatrix}
    0 \\
    0 \\
    Kq
\end{bmatrix} + h_g
\]  

(1)

with rigid body accelerations \(\nu_{IR}, \dot{\omega}_{IR}\), overall mass \(m\), inertia tensor \(J\), center of gravity \(c\), and generalized forces \(h_g\). The elastic coordinates \(q\) are coupled with the rigid body motion, described by the reference frame, by the coupling matrices \(C_i, C_r\).

The elasticity description of the system is a second order differential equation, e.g. provided by a linear finite-element (FE) model

\[
M\ddot{q} + Kq = h_e \quad \text{with} \quad M = M^T > 0, \quad K = K^T \geq 0, \quad M, K \in \mathbb{R}^{N \times N}.
\]  

(2)

The time-dependent elastic coordinates \(q\) represent the motion of the nodes of the FE mesh and \(h_e\) represents time-dependent forces acting on the body. Due to constantly growing demands on the accuracy of numerical simulations, the spatial discretization of FE models tends to become finer. The dense meshing leads to models that may contain up to millions of degrees of freedom, [2]. Implementing such an elasticity description into an EMBS and performing a time integration is not possible due to the massive numerical effort. There exist several approaches to reduce the number of elastic coordinates in order to decrease the numerical burden and make time integrations feasible for complex elastic systems.

3 MODEL ORDER REDUCTION FOR MECHANICAL SYSTEMS

The linear finite-element equation of motion (2) defines the system equation of a second order linear time-invariant (LTI) system as it is used in control theory. The force excitation is interpreted as input \(u \in \mathbb{R}^p\) by introducing an input matrix \(B \in \mathbb{R}^{N \times p}\) that distributes the forces to the application points. Displacements of interest are captured in the output \(y \in \mathbb{R}^q\) via an output matrix \(C \in \mathbb{R}^{q \times N}\)

The transfer matrix of the LTI system then reads

\[
H(s) = C(s^2M + K)^{-1}B, \quad H(s) \in \mathbb{C}^{q \times p}
\]  

(3)

with the complex Laplace variable \(s\). In order to allow the time-integration of the elastic body in acceptable time a model order reduction must be performed. In order to retain the physical meaning of mass and stiffness matrices, a structure preserving reduction is required. The number of independent degrees of freedom \(N\) shall be reduced to a much smaller number \(n \ll N\) while maintaining the important dynamical properties of the system. Therefore, a suitable subspace \(\mathcal{Y}\) is required that is spanned by the columns of the projection matrix \(V \in \mathbb{R}^{N \times n}\). The nodal displacements are approximated by the reduced generalized coordinates \(q_{\text{red}}\)

\[
q \approx Vq_{\text{red}}.
\]  

(4)

Using a Galerkin projection, the reduced system matrices read

\[
\{M_{\text{red}}, K_{\text{red}}\} = V^T \{M_{\text{red}}, K_{\text{red}}\} V, \quad \{M_{\text{red}}, K_{\text{red}}\} \in \mathbb{R}^{n \times n}, \\
B_{\text{red}} = V^T B, \quad B_{\text{red}} \in \mathbb{R}^{n \times p}, \quad C_{\text{red}} = CV, \quad C_{\text{red}} \in \mathbb{R}^{q \times n}.
\]  

(5)
The use of orthogonal projection ensures that symmetry properties are retained in the reduced system. The reduced transfer matrix, which is of the same dimension as for the original system, is

\[
H_{\text{red}}(s) = C_{\text{red}}(s^2M_{\text{red}} + K_{\text{red}})^{-1}B_{\text{red}}, \quad H_{\text{red}}(s) \in \mathbb{C}^{q \times p}.
\] (7)

There exist many different algorithms for the generation of the projection matrix \(V\). One possible classification of many of the available algorithms is the separation into interpolation-based and truncation-based methods.

### 3.1 Interpolation-Based Algorithms

The transfer function (3) can be expressed exactly with a power series

\[
H(s) = \sum_{j=0}^{\infty} -T_j^{\sigma_0}s^j = \sum_{j=0}^{\infty} -\frac{1}{j!} \frac{\partial^j H(s)}{\partial s^j} s^j
\] (8)

with the so-called moments \(T_j^{\sigma_0}\) of order \(j\) around the shift \(\sigma_0\). Moment matching is a procedure that ensures that certain moments of original and reduced system match and thus, the value and the derivatives of \(H(s)\) and \(H_{\text{red}}(s)\) up to a requested order at selected shifts are equal. A transfer function of a reduced model meeting these requirements is denoted a multipoint Padé approximant or a Hermite rational interpolant, [3]. Therefore, we call algorithms that generate reduced systems with matching moments interpolation-based. An explicit calculation of the moments is numerically unstable. However, moment matching can be guaranteed implicitly by using Krylov subspaces. The basis for Krylov subspaces can be obtained in a numerically robust manner by Arnoldi or Lanczos algorithms [3]. There exist Krylov subspace definitions for first and second-order systems [4, 5]. In case of a conservative mechanical system, the first-order Krylov subspace of order \(r\) can be used for moment matching

\[
\mathcal{X}_r(A_2,A_1) = \text{colspan}(A_1,A_2A_1,\ldots,A_2^{r-1}A_1) \subseteq \mathcal{Y} = \text{colspan}(V)
\] (9)

with \(A_1 = -(\sigma_0^2M + K)^{-1}B, \quad A_2 = -(\sigma_0^2M + K)^{-1}M\). (10)

Moment matching is closely related to static condensation [6] which is widely used as model reduction for mechanical systems, especially as part of the Craig-Bampton scheme. One can regard the static Guyan condensation as a special case with the moment of order \(j = 0\) around \(\sigma_0 = 0\) being matched, see [2] for details.

### 3.2 Truncation-Based Algorithms

Instead of using an interpolation of the transfer function, truncation-based methods can also be applied in order perform a reduction. Given the dynamical system in a certain representation, a large amount of the states of the system is simply neglected respectively truncated. For a system in its nodal representation, Equation (2), this obviously makes no sense for most applications. The key step in truncation methods is to find a system representation in which the importance of the single states for the dynamical properties is known or can be estimated. Usually, the system is transformed into a representation where certain invariants of the system are directly related to single states, i.e. eigenvalues or Hankel singular values (HSV). By then truncating states, the remaining invariants are maintained in the reduced system.

A typical and very well known method from this area is modal truncation. By solving the eigenvalue problem

\[
(\lambda^2M + K)\phi_i = 0 \quad \forall i = 1,\ldots,N;
\] (11)

we obtain the matrix of eigenmodes \(\Phi = [\phi_1,\ldots,\phi_N]\) which is used to transform the system into modal space. The new states of the system are then the eigenmodes respectively the modal amplitudes. Note that the eigenvalues of the reduced system are equal to the selected eigenvalues.
of the original system. Important eigenmodes, usually those related to low eigenfrequencies, are kept while the others are truncated. The determination of the importance of each eigenmode is, however, not always straightforward. Obviously, the loading situation of the system, which can have a large effect on the resulting deformations, is not considered in Equation (11). In practice, especially for large systems, only a certain amount of eigenmodes are calculated using an iterative eigensolver, e.g. [7], instead of performing a transformation followed by a truncation.

Another popular method is balanced truncation. The system is transformed into the so-called balanced realization and unimportant states which are identified by the system invariant HSV are then truncated. The HSV of the reduced system are then equal to the largest HSV of the original system. They are defined as the square roots of the spectrum of the product of Gramian controllability and observability matrix

$$\sigma_{HSV,i} = \sqrt{\lambda_i(PQ)},$$  \hfill (12)

see [8] for details. In the balanced realization the Gramian matrices of the system are equal, diagonal and contain the Hankel singular values

$$P = Q = \text{diag}(\sigma_{HSV,i}).$$  \hfill (13)

The states are sorted from easy to excite and easy to observe to hard to excite and hard to observe. In contrast to modal truncation, the loading conditions are explicitly considered. It is also possible to incorporate a frequency weighting into the algorithm. For second order systems, there exist multiple Gramian controllability and observability matrices, see [9, 10]. In the context of this contribution, we want to focus on orthogonal projection only in order to maintain symmetry. Since a stable system is required, we consider a viscous damping defined by the damping matrix $D$. As proposed by [10, 11, 12] we seek for the eigendecomposition of the Gramian controllability matrix on position level

$$P_p = \frac{1}{\pi} \int_{0}^{0_2} L^{-1}(\omega)BB^T L^{-H}(\omega)d\omega \quad \text{with} \quad L(\omega) = -\omega^2 M + i\omega D + K. \hfill (14)$$

The projection matrix $V$ is defined as the first eigenvectors of $P_p$

$$P_p V = \text{diag}(\zeta) V. \hfill (15)$$

If inputs and outputs are collocated, i.e. $C^T = B$, the eigenvalues $\zeta_i$ of $P_p$ are equal to the square roots of the spectrum of the product of Gramian controllability matrix on position level and Gramian observability matrix on position velocity level [10]

$$\zeta_i = \sqrt{\lambda_i(P_pQ_{pv})}, \hfill (16)$$

which are the second order analogy of the Hankel singular values. Please note that, similar to modal truncation, first the system is transformed into a certain realization and afterwards states that are considered unimportant are simply truncated. The difference to modal truncation is the chosen realization before the truncation. There exist several methods for the numerical evaluation respectively approximation, such as the ADI algorithms [13] or two-step approaches [14].

3.3 Component Mode Synthesis

Component mode synthesis methods usually combine different approaches for the reduction of an elastic structure, see [15] for an overview. There exist many variants which have been used in structural mechanics since the 1960’s, [16, 17, 18]. In the most prominent existing methods, interpolation-based modes like constraint modes (Craig-Bampton) or attachment modes (Rubin-McNeal) are combined with certain eigenmodes of the system. This type of combination of ansatz functions is known to deliver good reduced models, especially in the substructuring respectively...
EMBS context. It will be explained in the following that the reason for the good approximation of substructure-type systems is the usage of interpolation-based methods. The frequently used Craig-Bampton method uses constraint modes from static condensation for so-called boundary degrees of freedom, augmented with eigenmodes of the constrained system which approximate the internal dynamics.

Recently, it has been shown that the eigenmodes in the Craig-Bampton scheme can be replaced with arbitrary input-output based MOR methods by defining a new input matrix for the dynamics of the constrained structure, see [2]. The internal dynamics of the system are excited by inertia forces that stem from an acceleration of the boundary degrees of freedom. By using second-order balanced truncation instead of eigenmodes, the approximation of the overall dynamical system can be vastly improved as shown in [19].

4 MODEL ORDER REDUCTION FOR COUPLED MECHANICAL SYSTEMS

We consider a linear model of two mechanical systems that are connected at \( p_c \) degrees of freedom by force elements described with the matrix \( K_c \in \mathbb{R}^{p_c \times p_c} \). The matrices \( B_{c,k} \in \mathbb{R}^{N_k \times p_c} \) distribute the coupling degrees of freedom to the nodal coordinates of subsystem \( k \). Furthermore, we consider global inputs and outputs to the assembled system located at different nodes, captured in the matrices \( B_{i,k} \in \mathbb{R}^{N_k \times p_i} \), \( C_{i,k} \in \mathbb{R}^{N_k \times q_i} \). The resulting second order LTI system reads

\[
\begin{bmatrix}
M_1 & 0 \\
0 & M_2
\end{bmatrix}
\ddot{q} + \begin{bmatrix}
K_1 + B_{c,1}K_cB_{c,1}^T & -B_{c,1}K_cB_{c,2}^T \\
-B_{c,2}K_cB_{c,1}^T & K_2 + B_{c,2}K_cB_{c,2}^T
\end{bmatrix}
q = \begin{bmatrix}
B_{i,1} & 0 \\
0 & B_{i,2}
\end{bmatrix}
u,
\]

\[y = \begin{bmatrix} C_{i,1} & 0 \\
0 & C_{i,2} \end{bmatrix} q. \tag{17}\]

In [20], an error bound for interconnected systems is presented. Depending on the \( H_\infty \)-error of the uncoupled subsystems the bound guarantees a maximum \( H_\infty \)-error for the assembly. However, if the elements in the matrix that describe the connection between the systems are large, i.e. \( K_c \), the error bound is not applicable because certain prerequisites are not met.

In order to verify theoretical results a numerical model of two Euler-Bernoulli beams connected at the free end of each beam is considered. The model is shown in Figure 1. The connection is carried out with a six-degrees-of-freedom coupling expressed with \( K_c = 10^4 I \in \mathbb{R}^{6 \times 6} \). The selected global input nodes are also depicted in Figure 1 leading to \( p_{i,k} = 6 \). Global outputs are collocated.

![Figure 1. Numerical example: Two Euler-Bernoulli beams connected at the free end with bushing \( K_c \).](image)

For the assembled system the modified modal assurance criterion [21]

\[
m_{ij} = \frac{(\Phi_{\text{ref}}^H M \Phi_{\text{test},j})^2}{(\Phi_{\text{ref}}^H M \Phi_{\text{ref},i})(\Phi_{\text{test},j}^H M \Phi_{\text{test},j})} \tag{18}
\]

is evaluated for the reduced systems after a mass orthogonalization. Furthermore we consider the approximation of the transfer matrix, Equation (3), as an important measure of the quality of approximation. The considered absolute error is defined as Frobenius norm of the error system

\[
\epsilon_{\text{abs}}(s) = \|H(s) - H_{\text{red}}(s)\|_{\text{Fro}}. \tag{19}
\]
4.1 Block-Structure Preserving Model Order Reduction

Having the complete equations for the assembled system, one could directly apply one of the various reduction schemes to the assembled system and obtain a projection matrix \( V_{\text{sys}} \). However, in EMBS, each of the subsystems must be equipped with its own elasticity description. Thus, one can define the block-structure preserving projection matrix \( V_{\text{BSP}} \) by splitting the projection matrix for the overall system into

\[
V_{\text{sys}} = \begin{bmatrix} V_1 \\ V_2 \end{bmatrix}, \quad V_{\text{BSP}} = \begin{bmatrix} V_1 & 0 \\ 0 & V_2 \end{bmatrix},
\]

if \( V_k \) is regular [22]. The matrices \( V_k \) are bases for the deformations of the \( k \)-th elastic body and can be applied individually. Note that the space spanned by the columns of \( V_{\text{sys}} \) is also contained in \( V_{\text{BSP}} \), i.e. \( \text{colspan}(V_{\text{sys}}) \subset \text{colspan}(V_{\text{BSP}}) \). Typical properties of the reduction algorithm are therefore retained if \( V_k \) is applied to the subsystems.

4.2 Separate-Bases Model Order Reduction

In general, the equation of motion for the overall assembly will not be known during the reduction procedure. Especially if substructures are exchanged the equations and therefore the projection matrices lead to a projection matrix for the whole assembly

\[
V_{\text{SBR}} = \begin{bmatrix} V_{\text{SBR},1} & 0 \\ 0 & V_{\text{SBR},2} \end{bmatrix}.
\]

It will be shown in the following that interpolation-based methods have one decisive advantage over truncation-based methods in this setup. The moment matching properties are still fulfilled for the assembled system, even if the reduction is carried out on substructure level.

5 MOMENT MATCHING FOR COUPLED SYSTEMS

Consider the system-level projection matrix obtained with a moment-matching approach of order zero in block formulation for the shift \( \sigma_0 \)

\[
V_{\text{sys}} = \sigma_0^2 \begin{bmatrix} M_1 & 0 \\ 0 & M_2 \end{bmatrix} + \begin{bmatrix} K_1 + B_{c,1}K_cB_{c,1}^T & -B_{c,1}B_{c,2}^T \\ -B_{c,2}K_cB_{c,1}^T & K_2 + B_{c,2}K_cB_{c,2}^T \end{bmatrix}^{-1} \begin{bmatrix} B_{c,1} & 0 \\ 0 & B_{c,2} \end{bmatrix}.
\]

We can find a different representation of this equation using the Sherman-Morrison-Woodbury formula [23]

\[
(L + MJN^T)^{-1} = L^{-1} - L^{-1}M(J^{-1} + N^TL^{-1}M)^{-1}N^TL^{-1},
\]

with

\[
L = \begin{bmatrix} \sigma_0^2M_1 + K_1 & 0 \\ 0 & \sigma_0^2M_2 + K_2 \end{bmatrix},
\]

\[
MJN^T = \begin{bmatrix} B_{c,1}K_cB_{c,1}^T & -B_{c,1}B_{c,2}^T \\ -B_{c,2}K_cB_{c,1}^T & B_{c,2}K_cB_{c,2}^T \end{bmatrix} = \begin{bmatrix} 0 & B_{c,1} \\ B_{c,2} & 0 \end{bmatrix} \begin{bmatrix} K_c & 0 \\ 0 & K_c \end{bmatrix} \begin{bmatrix} -B_{c,1}^T & B_{c,2}^T \\ B_{c,1} & -B_{c,2} \end{bmatrix}.
\]
Using the abbreviations
\[ G_k = (O_0^2 M_k + K_k)^{-1}, \quad R_{i,k} = G_i B_{i,k}, \quad R_{c,k} = G_c B_{c,k}, \] (28)
the projection matrix from Equation (24) can be rewritten as
\[ V_{sys} = \begin{bmatrix} G_1 & 0 \\ 0 & G_2 \end{bmatrix} - \begin{bmatrix} 0 & 0 \\ 0 & B_{c,1} \end{bmatrix} \begin{bmatrix} K_{c,loc}^{-1} & 0 \\ 0 & K_{c,loc}^{-1} \end{bmatrix} + \begin{bmatrix} -B_{c,1}^T B_{c,2}^T & \begin{bmatrix} G_1 & 0 \\ 0 & G_2 \end{bmatrix} \end{bmatrix}^{-1} \begin{bmatrix} B_{c,1}^T \\ 0 \end{bmatrix} \] (29)

We can split the projection matrix for block-structure preservation according to the single components as in Equation (20)
\[ V_{BSP} = \begin{bmatrix} R_{i,1} + R_{c,1} \Gamma_1 & R_{c,1} \Gamma_2 & 0 & 0 \\ 0 & 0 & R_{c,2} \Gamma_3 & R_{c,2} + R_{c,2} \Gamma_4 \end{bmatrix}. \] (30)

Since we know that \( \text{colspan}(V_{sys}) \subset \text{colspan}(V_{BSP}) \), the moments at \( \sigma_0 \) are still matched using \( V_{BSP} \).

Applying the moment matching in a separate-bases approach for each uncoupled substructure, Equation (22), we obtain
\[ V_{SBR} = \begin{bmatrix} R_{i,1} & R_{c,1} & 0 & 0 \\ 0 & 0 & R_{c,2} & R_{r,2} \end{bmatrix}. \] (31)

Depending on the number of coupling degrees of freedom \( p_c \) and global inputs \( p_{i,k} \), there exist the following cases:

- \( p_{i,k} \geq p_c \forall k \Rightarrow \text{colspan}(V_{BSP}) = \text{colspan}(V_{SBR}) \)
  
  In this case the moments for the coupling degrees of freedom are also matched when using system level ansatz functions and splitting them according to Equation (20). All moment matching properties are retained performing the separate-bases reduction. In the case that \( p_{i,k} = p_c \forall k \) both \( V_{BSP}, V_{SBR} \in \mathbb{R}^{N_1+N_2 \times 2p_c} \). Please note that if \( \exists k, p_{i,k} > p_c \), \( V_{BSP} \in \mathbb{R}^{N_1+N_2 \times p_{i,k} + p_{c,2}} \) is a singular matrix.

- \( \exists k, p_{i,k} < p_c \Rightarrow \text{colspan}(V_{BSP}) \subset \text{colspan}(V_{SBR}) \)
  
  The moments for the coupling degrees of freedom are not matched when using system level ansatz functions and splitting them according to Equation (20). However, if \( p_{i,k} \leq p_c \forall k \), the column dimension of \( V_{BSP} \) is also smaller than \( V_{SBR} \) while matching the moments for the global input degrees of freedom. Also, if \( \exists k, p_{i,k} > p_c \), \( V_{BSP} \) is a singular matrix.

If global input degrees of freedom coincide with coupling degrees of freedom, both approaches will deliver rank deficient matrices in the presented setup. This can be dealt with by deflation strategies, e.g. incorporated in a modified Gram-Schmidt orthogonalization.

It is possible to derive the same results for rational interpolation of arbitrary order, compare [22]. The crux of the matter is that the guaranteed results, i.e. matching moments of the transfer function at selected shifts, are still valid if the reduction is performed on unconstrained single bodies that
are later assembled. In particular, no knowledge about the stiffness of the connection and the opponent interacting body are required during the reduction of the single component. Only the location of interaction needs to be provided.

Applying block-structure-preserving and separate-bases moment matching to the example system confirms the results. Each beam is reduced with 12 ansatz functions, according to the number of inputs, for the shift $\sigma_0 = 2\pi i 30$. Figure 2 shows the modal assurance criterion. All eigenmodes of the reduced systems coincide. The singular values of $[V_{BSP} V_{SBR}]$ show a clear drop and therefore indicate a rank deficiency of 24, i.e. the spaces spanned by the two projection matrices are equal.

The theoretic results are further verified in Figure 3. Moment matching is maintained even if performed on substructure level. However, the global approximation quality can be bad, especially in frequency ranges where no shifts are considered.

**Figure 2.** From left to right: modal assurance criterion (BSP versus SBR) and singular values of $[V_{BSP} V_{SBR}]$ for moment matching at $\sigma_0 = 2\pi i 30$.

**Figure 3.** Transfer function approximation error $\varepsilon_{\text{abs}}(f)$ for moment matching at $\sigma_0 = 2\pi i 30\text{Hz}$. 
6 TRUNCATION BASED MOR FOR COUPLED SYSTEMS

In contrast to the interpolation-based methods shown in the preceding section, results from the BSP approach differ from those obtained with the SBR approach for truncation methods. This means that the typical properties of the MOR scheme can not be guaranteed if it is applied to subsystems individually.

6.1 Modal Truncation

The $i$-th eigenmode for the coupled system (17) is defined by the eigenproblem

$$ \left( \lambda_i^2 \begin{bmatrix} M_1 & 0 \\ 0 & M_2 \end{bmatrix} + \begin{bmatrix} K_1 + B_{c,1}^T K_c B_{c,1}^T & -B_{c,1} K_c B_{c,2}^T \\ -B_{c,2} K_c B_{c,1}^T & K_2 + B_{c,2}^T K_c B_{c,2} \end{bmatrix} \right) V_{\text{sys},i} = 0. \quad (32) $$

The matrices $V_k$, that are needed for the block-structure preserving reduction, therefore fulfill

$$ \left( \lambda_i^2 M_1 + K_1 + B_{c,1}^T K_c B_{c,1}^T \right) V_{1,i} - B_{c,1} K_c B_{c,2}^T V_{2,i} = 0, \quad (33) $$

$$ \left( \lambda_i^2 M_2 + K_2 + B_{c,2}^T K_c B_{c,2} \right) V_{2,i} - B_{c,2} K_c B_{c,1}^T V_{1,i} = 0. \quad (34) $$

The submatrices for the separate-bases reduction are the solutions of the eigenproblems

$$ \left( \lambda_i^2 M_1 + K_1 \right) V_{\text{SBR},1,i} = 0, \quad (35) $$

$$ \left( \lambda_i^2 M_2 + K_2 \right) V_{\text{SBR},2,i} = 0. \quad (36) $$

The spaces spanned by $V_{\text{BSP}}$ and $V_{\text{SBR}}$ will be identical if

$$ B_{c,1} K_c B_{c,1}^T V_1 = B_{c,1} K_c B_{c,2}^T V_2, \quad (37) $$

$$ B_{c,2} K_c B_{c,2}^T V_2 = B_{c,2} K_c B_{c,1}^T V_1, \quad (38) $$

which is true if the systems are not coupled at all. Therefore, a modal truncation on substructure level delivers results that differ from an assembly-level modal truncation, i.e. the exact eigenmodes of the assembly are not included in $V_{\text{SBR}}$.

The MAC values and the singular values shown in Figure 4 clearly confirm that for modal truncation, the two approaches deliver different results. The higher eigenmodes of the assembled system are not captured with the separate bases approach. There is no clear drop in the singular values.

![Figure 4](image-url)
values and it is hard to determine if there is a rank deficiency at all. Obviously, \( \text{colspan}(V_{\text{BSP}}) \neq \text{colspan}(V_{\text{SBR}}) \) if modal truncation is considered.

For the simple example, 12 eigenmodes of the single structures do not suffice to approximate the transfer function in an acceptable quality, as can be observed in Figure 5. Even the first two eigenfrequencies are not approximated well. A straightforward application of modal truncation is therefore not well suited for the SBR approach.

Figure 5. Transfer function approximation error \( \epsilon_{\text{ab}}(f) \) for modal truncation.

### 6.2 Balanced Truncation

The second-order frequency-weighted controllability Gramian matrix on position level for the assembled system (17) is

\[
P_{p,\text{sys}} = \frac{1}{\pi} \int_{0}^{\infty} L_{\text{sys}}^{-1}(\omega)B_{\text{sys}}B_{\text{sys}}^T L_{\text{sys}}^{-H}(\omega) d\omega
\]  

with

\[
L_{\text{sys}}(\omega) = \begin{bmatrix} L_1 + B_{c,1}K_cB_{c,1}^T & -B_{c,1}K_cB_{c,2}^T \\ -B_{c,2}K_cB_{c,1}^T & L_2 + B_{c,2}K_cB_{c,2}^T \end{bmatrix},
\]  

\[
L_k = -\omega^2M_k + i\omegaD_k + K_k, \quad B_{\text{sys}} = \begin{bmatrix} B_{i,1} & 0 \\ 0 & B_{i,2} \end{bmatrix}.
\]  

The first \( n \) eigenvectors of \( P_{p,\text{sys}} \) are used as projection matrix \( V_{\text{sys}} \). The \( i \)-th eigenvector of the Gramian matrix is the solution of the eigenproblem

\[
(\xi_i^2 I + P_{p,\text{sys}}) V_{\text{sys},i} = 0.
\]  

For each uncoupled subsystem one can compute

\[
P_{p,k} = \frac{1}{\pi} \int_{0}^{\infty} L_k^{-1}(\omega)B_kB_k^T L_k^{-H}(\omega) d\omega \quad \text{with} \quad B_k = \begin{bmatrix} B_{i,k} & B_{c,k} \end{bmatrix}.
\]  

Using a separate-bases reduction leads to the definition of a modified Gramian matrix on system level

\[
P_{p,\text{SBR}} = \frac{1}{\pi} \int_{0}^{\infty} L_{\text{SBR}}^{-1}(\omega)B_{\text{SBR}}B_{\text{SBR}}^T L_{\text{SBR}}^{-H}(\omega) d\omega
\]  

with

\[
L_{\text{SBR}}(\omega) = \begin{bmatrix} L_1 & 0 \\ 0 & L_2 \end{bmatrix}, \quad B_{\text{SBR}} = \begin{bmatrix} B_{i,1} & B_{c,1} & 0 & 0 \\ 0 & B_{i,2} & B_{c,2} \end{bmatrix},
\]
which differs from the actual Gramian matrix for the interconnected system $P_{p,sys}$.

For the evaluation of the example system, a frequency weighting from 0 to 100Hz is considered. From the MAC values and the singular values of $[V_{BSP} V_{SBR}]$, Figure 6, we observe that the two approaches do not deliver the same reduced system. This is indeed similar to the result obtained with modal truncation.

Looking at the approximation of the transfer function, depicted in Figure 7, it can be observed that the SBR approach yields a worse approximation of the assembled system than the direct application or the BSP reduction. However, the overall approximation of the assembled system is the best of the presented approaches due to the nice convergence properties of balanced truncation, [24]. Particularly in comparison to modal truncation the improvement is immense. Nevertheless, a good approximation can not be guaranteed in general and is strongly dependent on the dynamics of the system including the connection and the reduction size, [2, 19].

**Figure 6.** From left to right: modal assurance criterion (BSP versus SBR) and singular values of $[V_{BSP} V_{SBR}]$ for second order balanced truncation with frequency weighting from 0 to 100Hz.

**Figure 7.** Transfer function approximation error $\varepsilon_{abs}(f)$ for second order balanced truncation with frequency weighting from 0 to 100Hz.
7 CONCLUSIONS

Elastic multibody systems consist of multiple bodies that may interact. It was shown that if single bodies are exchanged or coupling conditions are altered, interpolation-based model order reduction methods, i.e. static condensation and Krylov-subspace methods, still fulfill the moment matching properties for the overall system. Truncation-based methods, i.e. modal and balanced truncation, do not keep their properties on assembly level.

However, especially balanced truncation has some advantages compared to the interpolation methods. The size of the reduced system is not dependent of the number of interactions and nice convergence properties are typical.

In traditional component mode synthesis a mixture of both approaches is used. Craig-Bampton uses static condensation for the interface dynamics and eigenmodes of the internal dynamics, which are not influenced by a changing environment. A mixture of static condensation and balanced truncation is applied in [2]. This ensures the compatibility in terms of changing environmental settings and at the same time has superior convergence properties.

REFERENCES


Interpolation-based Parametric Model Order Reduction for Material Removal in Elastic Multibody Systems

Michael Fischer, Peter Eberhard

Institute of Engineering and Computational Mechanics
University of Stuttgart
Pfaffenwaldring 9, 70569 Stuttgart, Germany
{michael.fischer, peter.eberhard}@itm.uni-stuttgart.de

ABSTRACT
One essential step in the simulation of elastic multibody systems is the model order reduction of the elastic degrees of freedom. For simulations of material removal, the system matrices can be described as parameter dependent. In this contribution, parametric model reduction methods based on interpolation are applied for this type of problem. Thereby, the interpolation of the reduced system matrices, interpolation of projection matrices, interpolation of subspaces and the interpolation of the transfer functions are investigated. The advantages and disadvantages of these techniques, especially for the application in elastic multibody systems are carved out and illustrated for the manufacturing of a T-shaped workpiece which varies in its thickness due to material removal.

Keywords: Parametric Model Order Reduction, Elastic Multibody System, Material Removal, Interpolation.

1 INTRODUCTION
For many mechanical systems elastic deformations as well as rigid body motions are of interest. These systems are modeled as elastic multibody systems (EMBS). The large number of elastic degrees of freedom, due to fine spatial discretization, requires model order reduction (MOR), for which various linear MOR-techniques are available, see [1].

For an increasing number of applications, e.g. simulation of turning and milling processes, gearwheels, sliding components, and cranes, the system matrices cannot be considered as constant anymore. In recent years, parametric model order reduction (PMOR) methods have been developed to reduce parametric systems, see [2] for a good overview. Parametric model reduction methods are often divided in global and local approaches. In global PMOR only one representative projection matrix is determined by combining projection matrices for different sampling points. In contrast, the local PMOR methods determine a new reduced system by interpolation of locally reduced support systems. In this contribution, a parameter dependent T-shaped workpiece, which arises in simulations of material removal in EMBS, is investigated. Thereby all system matrices vary with the parameter, which defines the position of the cutting tool. In this contribution, four methods are investigated for the parameter-dependent elastic body. On the one side, the projection matrices are interpolated [3] and interpolated reduced systems are generated via projection. Related to this idea, the interpolation of the subspaces [4] is possible. Another approach is to interpolate the reduced system matrices [5, 6] which does not involve the original system in the online-interpolation-step anymore. This approach provides very good results for simulations of moving loads in EMBS, see [7, 8]. Last, the interpolation of the transfer function [9] enables a determination of one reduced system containing the parameter dependency.

The paper is structured as follows. First, the background about EMBS and the linear, non-parameter-dependent MOR is shortly presented in Section 2. The different parametric MOR techniques are discussed in Section 3. The description of the numerical model is introduced afterwards and the different PMOR-methods are compared for the T-shaped workpiece in Section 4 which is followed by the conclusion of this paper.
2 BACKGROUND

2.1 Elastic Multibody Systems

The elastic continuum is spatially discretized with the Finite Element Method which results in the ordinary differential equation

\[ M \cdot \ddot{q}(t) + D \cdot \dot{q}(t) + K \cdot q(t) = h_c(t) \]  

(1)

with the nodal displacements \( q(t) \in \mathbb{R}^N \), the positive definite mass-, stiffness and velocity-proportional damping matrices \( M, K, D \in \mathbb{R}^{N \times N} \). The acting forces on the elastic structure are denoted by \( h_c(t) \). The resulting elastic body is used in the floating frame of reference environment in [10] to describe the linear deformation in the nonlinear differential equation

\[
\begin{bmatrix}
    mE & \cdot m\tilde{c}^T(q) & C_C^T(q) \\
    m\tilde{c}(q) & J(q) & C_C^T(q) \\
    C_C(q) & C_C & M
\end{bmatrix} \cdot \begin{bmatrix} a(t) \\ \alpha(t) \\ \dot{q}(t) \end{bmatrix} = \begin{bmatrix} h_l(t) \\ h_r(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -K \cdot q(t) - D \cdot \dot{q}(t) \end{bmatrix}
\]  

(2)

for an elastic body with the rigid-body translational and rotational acceleration \( a, \alpha \), the mass \( m \), inertia \( J \), center of mass \( \tilde{c} \) and the acting forces \( h_l, h_r \). The rigid and elastic parts of (2) are coupled by \( C_C(q) \) and \( C_C \), which depend on the elastic deformation. See [10] for a detailed description of the generation of necessary data for a complete elastic multibody system.

2.2 Model Order Reduction in EMBS

To enable efficient EMBS simulations, the number of elastic degrees of freedom is reduced. Thereby, the elastic body is cut free and the resulting reaction forces are represented by inputs \( u(t) \), distributed by the input matrix \( B \in \mathbb{R}^{N \times r} \), in the linear time invariant system

\[ M \cdot \ddot{q}(t) + D \cdot \dot{q}(t) + K \cdot q(t) = B \cdot u(t), \]  

\[ y(t) = C \cdot q(t). \]  

(3)

The outputs of the system \( y(t) \) are determined by the nodal displacements of interest, calculated with the output matrix \( C \in \mathbb{R}^{o \times N} \). The goal in model order reduction is the replacement of this equation of motion with a reduced-order model of lower dimension \( n \ll N \) which still represents the decisive dynamics and could be used cheaply and effectively in EMBS. Therefore, the structure of the system matrices cannot be modified and structure-preserving model order reduction by a Petrov-Galerkin projection is applied. The elastic coordinates \( q \approx V \cdot \bar{q} \) are approximated by the reduced coordinates \( \bar{q} \in \mathbb{R}^n \) in the subspace \( \mathcal{Y} \) spanned by the columns of the projection matrix \( V \in \mathbb{R}^{N \times n} \). After plugging this approximation into (3) and enforcing orthogonality of the residual to \( \mathcal{W} = \text{span}(W) \) the equation of motion of the reduced body reads

\[ W^T \cdot M \cdot V \cdot \ddot{\bar{q}}(t) + W^T \cdot D \cdot V \cdot \dot{\bar{q}}(t) + W^T \cdot K \cdot V \cdot \bar{q}(t) = W^T \cdot B \cdot u(t), \]  

\[ \bar{y}(t) = C \cdot V \cdot \bar{q}(t). \]  

(4)

In this contribution, only orthogonal projection with \( W = V \) is applied.

The transfer function of a linear second order system, original as well as reduced system, \( H(s) = C \cdot (s^2 M + s D + K)^{-1} \cdot B \) is used to describe the dynamics of the system in the frequency domain. To evaluate the approximation quality of the reduced systems, the relative error

\[ \varepsilon(i\omega) = \frac{\|H(i\omega) - \hat{H}(i\omega)\|_F}{\|H(i\omega)\|_F} \]  

(5)

is defined frequency wise, which correlates to the \( \mathcal{H}_2 \)-error norm defined in [1].
2.3 Linear Model Order Reduction

The major idea of the Component Mode Synthesis (CMS) is based on the sub-structuring of Finite Element structures, [11]. Eigenmodes $\Phi_{\text{eig}}$ for proportionally-damped systems, which are calculated by the conservative eigenvalue problem

$$ (\lambda_{i}^2 M + K) \cdot \Phi_{\text{eig},i} = 0, \quad i = 1, \ldots, N $$

(6)

often cannot provide reduced systems meaningful for EMBS simulations. Therefore, correction modes, like attachment or static modes $\Phi_{\text{SM}} \in \mathbb{R}^{N \times r}$ for constrained systems

$$ K \cdot \Phi_{\text{SM}} = B $$

(7)

are used to extend the modal reduction based projection matrix

$$ V_{\text{CMS}} = [\Phi_{\text{eig},i} \Phi_{\text{SM}}] \in \mathbb{R}^{N \times n}, \quad i = 1, \ldots, n - r. $$

(8)

The size of the projection matrix $V$ depends on the number of columns $r$ of the input matrix $B$ and the number of used eigenmodes $n - r$. In [11] further correction modes and modes for systems with rigid body degrees of freedom are discussed.

It is the basic moment matching idea to match certain coefficients of the series expansion of the original and reduced transfer functions $H(i\omega)$ and $\bar{H}(i\omega)$. Depending on the number of matched moments, which represents the number of coefficients, the reduced system approximates the original system at the chosen expansion points $\sigma_i$. If the value of the transfer functions and their derivatives

$$ H(\sigma_i) = \bar{H}(\sigma_i), \quad \frac{\partial H}{\partial \sigma}(\sigma_i) = \frac{\partial \bar{H}}{\partial \sigma}(\sigma_i), \quad i = 1, \ldots, l $$

(9)

should coincide for $l$ expansion points $\sigma_i$, the projection matrix is calculated by

$$ \mathcal{V} = \text{span}[(\sigma_1^2 M + \sigma_1 D + K)^{-1} \cdot B, \ldots, (\sigma_l^2 M + \sigma_1 D + K)^{-1} \cdot B], $$

$$ \mathcal{W} = \text{span}[(C \cdot (\sigma_1^2 M + \sigma_1 D + K)^{-1})^T, \ldots, (C \cdot (\sigma_l^2 M + \sigma_1 D + K)^{-1})^T]. $$

(10)

This multi-moment matching can lead to reduced systems which well approximate the original system in a wide frequency range. A detailed description for higher orders, about Krylov subspaces and the Arnoldi algorithm to calculate the projection matrices efficiently in EMBS can be found in [12, 13].

3 PARAMETRIC MODEL ORDER REDUCTION

3.1 General

In the previous section the system matrices of the dynamical system in (3) are constant. In a rising number of applications, the system matrices are not constant anymore and, therefore, the system matrices in the equation of motion

$$ M(p) \cdot \ddot{q}(t) + D(p) \cdot \dot{q}(t) + K(p) \cdot q(t) = B(p) \cdot u(t), $$

$$ y(t) = C(p) \cdot q(t) $$

(11)

depend on parameters $p = [p_1, \cdots, p_d]$. In this contribution, the parameter represents the position of the cutting tool and is described by a scalar value $p = p_1$. Equation (11) is still linear in its state, but it can be meaningful that all system matrices contain a nonlinear parameter dependency.

One goal in parametric model order reduction is the maintenance of the parameter dependency in the reduced system, which is illustrated in Figure 1 for second order mechanical systems. Projection-based model reduction is nicely summarized from the mathematical perspective in [2] and the classification of the various methods in this paper is similar to it.
This PMOR-techniques can be separated in two classes, global and local PMOR-methods. Both consist of individual calculations of reduced systems for certain parameter samples \( p_1, \ldots, p_k \). Afterwards, the generated reduced systems, represented by the individual projection matrices \( V_1, \ldots, V_k \), have to be combined in a reasonable way. In global PMOR these matrices are combined to a global projection matrix \( V = [V_1, \ldots, V_k] \). In previous work of the authors, see [7], the necessity to include many support systems to achieve satisfying results was carved out. For global PMOR the size of the generated reduced system depends on the number of parameter samples. This curse of dimensionality represents the largest disadvantage of this method. In contrast, local PMOR methods enable an interpolation-based calculation of systems for any parameter value depending on the generated support systems. In the following, four methods are presented and applied for the parametric model reduction of a mechanical system.

### 3.2 Interpolation of Projection Matrices

The direct interpolation between the offline-calculated individual projection matrices \( V_1, \ldots, V_k \) for sampled parameter values is not always meaningful and the interpolated system matrices might lose decisive properties. Therefore, in [3] the interpolation of the projection matrices on a tangent space to a manifold of the subspaces spanned by the columns of the projection matrices \( V_i \in S_i \) is proposed. For this purpose, the tangent space \( T_r \mathcal{M} \) of the manifold \( \mathcal{M} \), at a reference point with the parameter value \( p_r \) is defined, as illustrated in Figure 2.

The other projection matrices \( V_i, i = 1, \ldots, k \) are mapped onto \( T_r \mathcal{M} \) by the logarithmic map

\[
T_i = \text{Log}_{V_r}(V_i) \in T_r \mathcal{M},
\]

which is represented by the dashed arrows in Figure 2. This mapping is done numerically with the
Singular Value Decomposition (SVD) of
\[
(I - V_r \cdot V_i^T) \cdot V_i \cdot (V_i^T \cdot V_i)^{-1} = U_i \cdot \Sigma_i \cdot Z_i^T \hspace{1em} \text{and} \hspace{1em} T_i = U \cdot \arctan(\Sigma_i)Z_i^T.
\]

This mapping represents a geodesic on the tangent manifold from $V_r$ to $V_i$. The matrices $T_i, i = 1, \ldots, k$ are interpolated $T_d = \sum_{i=1}^{k} \omega(p_d)T_i$ for a parameter value of interest $p_d$. Afterwards, this matrix is used for mapping back onto the original subspace via the exponential map
\[
V_d = \text{Exp}_{V_i}(T_d).
\]

This is executed by the thin SVD of $T_d$
\[
T_d = U_d \cdot \arctan(\Sigma_d)Z_d^T \hspace{1em} \text{and} \hspace{1em} V_d = V_r \cdot Z_d \cdot \cos(\Sigma_d) + U_d \cdot \sin(\Sigma_d).
\]

The calculated projection matrix $V_d$ for a certain parameter value $p_d$ is applied in the projection framework to provide reduced system matrices. Thereby, the original system for the parameter value $p_d$ has to be provided.

### 3.3 Interpolation of Subspaces

Similar to the idea of interpolating between the projection matrices, in [4] a method for interpolation between subspaces is proposed, which is explained briefly here. It is obvious that, although the subspaces of the individual reduced systems $S_i$ might change slightly for small parameter changes, the columns of the projections matrices do not have to show this behavior. This is illustrated exemplary in Figure 3 where the subspaces $S_1, S_2$ spanned by $V_1$ and $V_2$ differ slightly but the projection matrices distinguish drastically.

First, the columns of the individual projection matrices $V_i$ are transformed. Therefore, the orthogonal projection $\xi_{S_i}$ of a reference basis $V_r$ onto the subspace $S_i = \text{range}(V_i)$ is applied. After this additional projection, the columns of the projection matrix are normalized by a scaling function $\theta_{i,r}$ which leads to the interpolated projection matrix
\[
V_d = V(p_d) = \sum_{i=1}^{k} \omega(p_d)\theta_{i,r} \circ \xi_{S_i}(V_r).
\]

With the matrix notation of the scaling function $\theta_{i,r}$
\[
e_i \cdot D_{i,r} \cdot e_i = ||V_i^T \cdot V_r \cdot e_i||^{-1}
\]

with the diagonal matrix $D_{i,r}$ and the $i$-th unit vector, the normalization and projection reads
\[
\theta_{i,r} \circ \xi_{S_i}(V_r) = V_i \cdot V_i^T \cdot V_r \cdot D_{i,r}.
\]

![Figure 3: Relation between subspace and projection matrix in interpolation of subspaces](image-url)
3.4 Interpolation of System Matrices

Here, it is the basic idea to apply the individual projection matrices \( V_1, \ldots, V_k \) to project the individual system matrices \( M(p_i), D(p_i), K(p_i), B(p_i), C(p_i) \) for \( i = 1, \ldots, k \) and interpolate between the resulting reduced system matrices. The direct interpolation between these matrices is not necessarily meaningful because the reduced coordinates \( \vec{q} \) can lie in different subspaces, and, therefore, have different meanings. The independently calculated reduced coordinates have to be transformed \( \vec{q} = T_j \cdot \vec{q}^j \). If orthogonal projection with \( V_i = W_i \) is applied, the resulting reduced system is written as

\[
\begin{align*}
    \bar{T}_j^T \cdot M_j \cdot T_j \cdot \vec{q}^j(t) + \bar{T}_j^T \cdot D_j \cdot T_j \cdot \vec{q}^j(t) + \bar{T}_j^T \cdot K_j \cdot T_j \cdot \vec{q}^j(t) &= \bar{T}_j^T \cdot B_j \cdot u(t), \\
    \bar{y}(t) &= \bar{C}_j \cdot T_j \cdot \vec{q}^j(t)
\end{align*}
\]

with the projection matrix \( \bar{V}_i = V_i \cdot T_i \) used to generate the reduced system matrices.

Two different approaches [5, 6] determine the transformation matrix differently. In [5], the individual transformation matrix

\[
T_i = (R^T \cdot V_i)^{-1} \quad \text{and} \quad R^T \cdot R = I
\]

is build with the matrix \( R \) which corresponds to the \( n \) most important directions of \( V_{all} \) with an SVD. This projection matrix is either calculated by \( V_{all} = [V_1 \ldots V_k] \) or a weighted combination \( V_{all} = [\omega_1(p_d) V_1 \ldots \omega_k(p_d) V_k] \) is applied. Afterwards, the direct interpolation between the transformed reduced system matrices

\[
\{\bar{M}(p_d), \bar{D}(p_d), \bar{K}(p_d), \bar{B}(p_d), \bar{C}(p_d)\} = \sum_{i=1}^k \omega_i(p_d) \{M_i, D_i, K_i, B_i, C_i\}
\]

is meaningful.

The interpolation on manifold sets in [6] requires a congruence transformation to express the individually reduced systems in consistent sets of generalized coordinates \( \vec{q}^* \). The goal is to define the transformation matrix \( T_i \) to minimize the difference between the transformed individual projection matrices \( V_i \cdot T_i \) and a projection matrix \( V_r \) at the parameter value \( p_r \), which represents a reference configuration

\[
\min_{T_i, \text{orthogonal}} \| V_i \cdot T_i - V_r \|^2_F. \tag{22}
\]

This minimization problem is equivalent to

\[
\max_{T_i, \text{orthogonal}} \text{trace}(T_i^T \cdot V_i^T \cdot V_r), \quad \text{with} \quad P_{i,r} = V(p_i)^T \cdot V(p_r). \tag{23}
\]

An analytical solution of this problem is given by the SVD of \( P_{i,r} \)

\[
P_{i,r} = U_i \cdot \Sigma_i \cdot N_i^T \quad \text{with} \quad T_i = U_i \cdot N_i. \tag{24}
\]

In [5], it depends on the interpolation scheme and it is not guaranteed that the interpolated reduced system matrices provide the same properties as the original system matrices, e.g. positive definiteness. Therefore, the interpolation on matrix manifolds is proposed in [6] and the transformed system matrices in (19) are not interpolated directly. With a logarithmic mapping \( \Gamma = \log_X(Y) \), the matrix \( Y \) is mapped to the tangent space \( T_X \mathcal{M} \) which is calculated at the reference configuration \( X \). On this tangential manifold the independent interpolation of the elements of the matrix \( \Gamma_d \) for a certain parameter value \( p_d \) is executed. The interpolated matrix \( \Gamma_d \) is subsequently mapped from the tangential manifold back to the original manifold by the exponential mapping \( \bar{Y}_d = \exp_X(\Gamma_d) \) which guarantees to determine interpolated matrices on the same manifold as the matrices used as support systems similar to the interpolation of the projection matrices in Figure 2.
3.5 Interpolation of Transfer Function

This local PMOR-approach bases on the combination of all individually reduced systems to one medium-sized model. The parameter dependency can be distributed in one of the resulting reduced system matrices due to the linearity of the transfer function. The interpolated reduced transfer function

$$\tilde{H}(s, p_d) = \sum_{i=1}^{k} \omega_i(p_d) H(s, p_i)$$

(25)

is calculated by the weighted sum of the reduced transfer function determined for the support systems. This includes the interpolation condition that the transfer function matches exactly if the parameter value $p_d$ is equal to a parameter value $p_i$ in the support system set. In [9] the construction of the interpolated system is proposed and should be explained here for second order systems similarly. The sum in (25) can be represented for any parameter value $p_d$ by

$$\sum_{i=1}^{k} \omega_i(p_d) H(s, p_i) = \tilde{C}(p_d) \cdot (s^2 \tilde{M} + s \tilde{D} + \tilde{K})^{-1} \cdot \tilde{B},$$

(26)

with

$$\tilde{M} = \text{diag}(\bar{M}(p_1), \ldots, \bar{M}(p_k)) \in \mathbb{R}^{m \times m}, \quad \tilde{D} = \text{diag}(\bar{D}(p_1), \ldots, \bar{D}(p_k)) \in \mathbb{R}^{m \times m},$$

$$\tilde{K} = \text{diag}(\bar{K}(p_1), \ldots, \bar{K}(p_k)) \in \mathbb{R}^{m \times m}, \quad \bar{B} = \begin{bmatrix} \tilde{B}(p_1) \\ \vdots \\ \tilde{B}(p_k) \end{bmatrix} \in \mathbb{R}^{m \times r},$$

$$\tilde{C} = [\omega_1(p_d) \tilde{C}(p_1), \ldots, \omega_k(p_d) \tilde{C}(p_k)] \in \mathbb{R}^{l \times m} \text{ with } m = n_1 + \cdots + n_k.$$ 

After the generation of the reduced system, the original matrices are not of interest anymore which makes this method suitable for non-affine parameter dependencies, too. The size of the reduced system depends on the number of support systems $k$ which are used in the offline-step. This curse of dimensionality represents the largest disadvantage of this method. In contrast to the other local PMOR-methods, this technique is the only one, which allows the splitting of the approximation error into a reduction and an interpolation error. Therefore, in [9] error bounds for the parametric system with balanced truncation are presented. Another disadvantage is the fact that the interpolated reduced system cannot represent other eigenfrequencies than the individually reduced support systems contain. For many examples, the eigenfrequencies vary with parameter changes which cannot be covered by the interpolation of the transfer function.

4 INVESTIGATED MODEL AND RESULTS

We consider the T-shaped, elastic workpiece shown in Figure 4. It is made of aluminum and the geometry is chosen to mimic typical lightweight parts like turbine blades or frame components. During machining, the cutter continuously removes material from the side of the plate along a part $h_m$ of its height. The thickness of the plate is reduced from $d$ to $d_m$. The position of the cutter and thus the machined length $l_m$ varies from $l_m = 0$ in the unmachined case to $l_m = l$ for the fully machined workpiece. A Finite Element routine is available, that meshes the plate using a regular spaced grid which leads to 69 models over the length $l$. The nodes on the lower part of the foot are constrained.

To investigate the quality and applicability of the different local PMOR-methods, the parametric T-shaped workpiece model is sampled from model 40 ($p = 0$) to model 60 ($p = 1$) to represent a part of the material removal process and limit the amount of data in this paper.

The quality of the different parametric model reduction techniques is evaluated regarding the relative error in frequency domain, see (5), for parameter values $p = 0.55$. Additionally, the relative static error $e_{\text{stat}}(p)$ for $f = 0$ for all parameters is investigated and the variation of the first five
Figure 4: Milling of an elastic, T-shaped workpiece

eigenfrequencies, which lie in the frequency range of interest, are calculated. These eigenfre-
cyency variations are summarized relatively in $\varepsilon_{\text{eig}}(p)$.

Interpolated systems are generated for the interpolation parameter values $p_{\text{int}} = 0.05(0.1)0.95$ with the support system set $p_s = 0(0.1)1$. This discretization results in 11 support systems and 10 interpolated systems which enables a more general statement about the quality of the different approaches.

4.1 Individual Model Reduction

Figure 5 shows the individual reduction error for the system with $p = 0.55$, which will be investi-
gated in the interpolation step, exemplary.

The modally reduced system shows the worst approximation because no input information is ap-
plied in the determination of the subspace. In contrast, in the CMS- and Krylov-based reduction
the static behavior is considered, which leads to exact results for $f = 0$. The Krylov reduction also
matches the transfer function at additional frequencies which provides very good results around
these points.
4.2 Interpolation of Projection Matrices

The support system, where the tangential space is determined, is fixed for $p_r = 0.5$ near the interpolation parameter value of interest $p = 0.55$. This should guarantee that the mapping onto the tangential manifold does not include a large error. The transfer function of the original and linear interpolated system for $p = 0.55$ is plotted in Figure 6.

The interpolation of the projection matrices provides very poor results, for modally as well as the input-output-based reduction techniques. The static error and the sum of the deviation of the eigenfrequencies, which are depicted in Figure 6 for the interpolation parameter space $p = 0.05(0.1)0.95$, are not of acceptable quality. Even for the modal reduction, the maximal relative difference of the eigenfrequencies amounts to 24.9 %. Although the original model at the parameter values of interest are used in the projection-step, a projection-matrix-based interpolation cannot provide satisfying results for an elastic body with varying material removal.

![Figure 6: Transfer function and interpolation error in interpolation of projection matrices](image)

4.3 Interpolation of Subspaces

The transfer function of the reduced systems generated by the interpolation of the subspaces show a similar behavior in Figure 7. This indicates that the error in the interpolation step exceeds the individual reduction error. Especially the eigenfrequencies are approximated for all three reduction methods with the same inaccuracy. Over the parameter space this deviation and the static error in Figure 7 for the reduction methods is similar, too. The average error of the static behavior is on the level of more than 10 %. The average of the eigenfrequency deviation is around 3 %.

![Figure 7: Transfer function and interpolation error in interpolation of subspaces](image)
4.4 Interpolation of Reduced System Matrices

In [7, 8], the interpolation based on the reduced system matrices is investigated for an elastic multibody system which is used to simulate a moving force around a thin-walled cylinder. Many specific characteristics have been investigated and a method to use this PMOR-method for simulation of moving loads was proposed. Here, the characteristics of the T-shaped workpiece are regarded and the major results are presented. See [7] for detailed explanations about the different problems and special features.

First, the trend of representative elements of the system matrices is illustrated in Figure 8 for both methods [5, 6]. Pay attention that the reduced system matrices in the approach in [6] cannot be interpolated directly but are mapped to the tangential manifold additionally.

The smaller the variation of the projection matrices $V_i$, the smoother the elements of the system matrices. Therefore, the modally reduced system is best suited if only the interpolation is regarded. As explained in [7], this error does not solely determine the approximation error between the original system and the interpolated reduced one. Therefore, the CMS-based reduction provides better results in Figure 8 for $p = 0.55$. The Krylov reduction is not suitable for this task because all columns of the projection matrices vary with the parameter value of the support systems and, therefore, the matrix elements change significantly which aggravates the interpolation, see [7] for detailed explanations.

The interpolation error for the modally reduced system are for both matrix interpolation techniques similar due to the high individual reduction error. The interpolated CMS-based reduced systems show the best results. A general statement which matrix interpolation method is better, cannot be provided based on this investigation. Therefore, the static and eigenfrequency error over the parameter space is calculated, see Figure 9. The approach in [6] provides the smallest static error but contains an increased error in the eigenfrequencies in comparison to the method in [5]. The interpolation of modally reduced systems approximate the eigenfrequencies very well but cannot describe the static behavior. For many mechanical systems, e.g. the turning of a thin-walled cylinder in [7], this difference in the static approximation quality can even increase if the eigenmodes cannot reflect the static behavior, especially for systems with rigid body degrees of freedom. Here, both matrix interpolation techniques provide very satisfying results, although the original system is not considered for the determination of the interpolated system in the parameter set $p_{int}$. This represents the largest benefit of this method because all matrices for online-calculations only depend on the size of the individually reduced systems.

Another large advantage is the fact that, for individual projection matrices which change smoothly
with parameter variations, high-order interpolation techniques might improve the interpolation results. This is shown exemplary for cubic-spline interpolations in Figure 9. Especially the approach in [6] benefits from higher-order interpolations. In contrast to the method proposed in [5], the type of interpolation cannot influence the characteristics of the matrices, e.g. positive-definiteness, because the interpolation is always executed on the tangential manifold and mapped backed to the original manifold. As investigated in [7], the interpolation quality does not have to improve automatically for higher-order interpolation methods, as described here for cubic splines. As the matrix elements might change more drastically for parameter variations, the abrupt changes can influence neighboring interpolation points.

4.5 Interpolation of Transfer Function

For the parameter values of the sample set all systems are reduced to the same size, although, in contrast to the other methods, this is not necessary in general.

Figure 10 shows the transfer function of the original and interpolated reduced system for the three different reduction methods. The eigenfrequencies are represented well for this example, but this is not characteristic for the interpolation method. Actually, the parameter sample grid is densely meshed which results in eigenfrequency deviations of less than 1% for neighboring parameter values. In contrast, the other values in the frequency range, especially the static behavior highly depends on the chosen input and the interpolation of the transfer function does not provide good

![Figure 9: Static and eigenfrequency error in interpolation of reduced system matrices](image1)

![Figure 10: Transfer function and interpolation error in interpolation of transfer function](image2)
results. Especially for the modally reduced system, for certain parameter values, as it is depicted for \( p = 0.55 \), the interpolation delivers very poor results. The input-output-based methods do not show this problem. The comparison of the three reduction methods in the interpolation over the parameter space in Figure 10 illustrates the large benefit of input-output-based reduction methods. The static error is extremely low for the CMS-based reduction which is based on the fine meshed parameter grid and the smooth changes in the transfer function for these parameter values. As described above, the eigenfrequency error is similar for all reduction methods and not shown here.

4.6 Comparison of Local PMOR-Methods

The relative error in frequency domain for the different PMOR-methods is summarized in Figure 11. Thereby, apart from the matrix interpolation, linear interpolation techniques are applied because high-order interpolation methods do not provide improved results. The only suitable solution for the problem of material removal for \( p = 0.55 \) is the interpolation of the reduced system matrices where both investigated methods provide satisfying results and even can deliver better results with high-order interpolation techniques.

![Figure 11: Comparison of different local PMOR-methods with CMS-based individual reduction](image)

5 CONCLUSIONS

The description of elastic multibody systems with material removal as parameter dependent systems and their reduction with interpolation-based parametric model order reduction is presented. Four methods, the interpolation of the projection matrices, system matrices, subspaces and transfer function were investigated for the numerical model of a T-shaped workpiece with material removal. Especially the PMOR-methods based on the interpolation of the reduced system matrices provide very satisfying results. This beneficial application is limited for individual reduction methods which generate projection matrices which change smoothly over a large parameter range. The CMS-based reduction and a linear interpolation already show very good results and can even be improved with high-order interpolation schemes. Depending on the application, the other PMOR-methods can be applied, too, however the interpolation of the transfer function can deliver small errors for densely meshed parameter grids but are not well suited for EMBS simulations.

REFERENCES


**Investigation on the modeling of membranes in large scale multi-body systems using co-rotational formulation**

**Jiabei Shi, Zhuyong Liu, Jiazhen Hong**

School of Naval Architecture, Ocean & Civil Engineering  
Shanghai Jiao Tong University  
800 Dongchuan RD. Minhang District, Shanghai, 200240, China  
sjbnust@163.com, zhuyongliu@sjtu.edu.cn, jzhong@sjtu.edu.cn

**ABSTRACT**

In large scale multi-body systems, such as solar arrays, there are always many deployable modules, which consist of thin and slim structures that may be beams, cables, shells or membranes. These structures make systems deploy easily and safely compared to complex mechanisms. On the contrary, it is much more complicated to simulate the response of them due to the large rotation and displacement in one structure. Furthermore, the most troublesome problem should be handled is the efficiency of the intensive computation.

In recent years, studies on large deformation (rotation and displacement) focus on Geometrically Nonlinear Theory, Absolute Nodal Coordinate Formulation and so on. Almost all the formulations are classified as typical Total Lagrangian description, and a large number of articles show that an accurate result relies on a strongly nonlinear internal force and its Jacobian, which should be computed for every step, even every iteration. Therefore, these articles present fine answers to the question that is simply small scale system with few beams or shells, but for a large scale system, most of methods do not work well for the intensive computation of stiffness matrix, internal force, etc.

In this paper, a co-rotational method is used to generate a shell model with a rotation-free element. In traditional co-rotational formulation, the rotation degree of freedom is introduced to express the bending strain, meanwhile, the rotation variable makes the computation complex and increases the degrees of freedom of the overall system. A rotation-free element in co-rotational method simplifies the internal force and avoids the singularity of rotation, and it also fits the co-rotational formulation for its simpler local bending strain expression compared to computing the local rotation in co-rotational coordinate frame. Moreover, a quite straightforward internal force matrix can be assembled with co-rotational formulation, and which is element-independent, hence, different elements lead to a similar high efficiency. The Time-Complexity and Space-Complexity analyses show a co-rotational description is good at solving a large scale system, which is verified by a solar array case.

**Keywords:** Co-rotational formulation, rotation-free, large scale, Shell.

1 **INTRODUCTION**

In recent years, the geometrical non-linear theory is relatively developed perfectly, and they settled many kinds of engineering problem focus on the Aerospace. The first try of solving non-linear structure is the geometrical non-linear theory, which is based on the finite rotation theory and objective strain description that make the configuration "geometrical accuracy". Based on the theory, lots of description of surface normal or section orientation are proposed for specific reasons, the most typical formulation of description is ANCF, which introduces the tangent vector as the only rotation variable that eliminates singularity of variables.

Currently, no matter which method, the troublesome problem is not the accuracy, but the efficiency. For the geometrical non-linear theory, the accurate result relies on a strongly nonlinear internal force and its Jacobian, which should be computed for every step, even every

This research work is supported by the National Science Foundation of China (11132007, 11202126), for which authors are grateful.
iteration. For the ANCF, the nodal degrees of freedom are huge. It cannot obtain a fast simulation for a practical problem without High Performance Computer.

Co-rotational (CR) was first used by Belytschko, who introduced element rigid convected frame for modeling beam type element and shell element using an explicit solution procedure [1], which was developed perfectly and the Commercial Code LS-Dyna was partly based on his contribution. The idea of co-rotational frame was introduced by Horrigmoe and Bergan [2,3], but the tangent stiffness was not directly used in iteration. Crisfield developed the concept of consistent Co-rotational formulation, where the stiffness matrix was used in solving nonlinear equations [4]. Ranking and Brogan introduced the concept of element independent Co-rotational formulation, the formulation used the projection matrix to extract the deformation variable from the overall variable, it was further refined by Ranking, Nour-Omid [5,6,7], and became a nonlinear shell analysis Code STAGS. Nowadays, Battini and coworkers used Co-rotational approach on stability applications.

For a set of problems, the displacements and rotations may be arbitrarily large, but deformation must be small [8]. In the other word, for a beam type or shell type structure, the displacements and rotations, maybe the deflection, experience a large variation, but the material experience a elastic deformation, the constitutive relation is constant. From the stated above, geometrical nonlinear problem always have a large displacement, but the most part of which is rigid body displacement. So the objective internal energy is contributed by the deformational displacement. From this point of view, CR approach is much more like the floating frame of reference formulation rather than the non-linear FEM, though CR is to solve geometrical non-linear problem. For these reasons, a new way to solve large scale problem is presented. The efficiency is primary consideration, meanwhile, the accuracy at some condition can be confirmed.

In this paper, a general co-rotational approach is restated by view of floating frame of reference formulation. In second section a rotation-free shell element is used in co-rotational formulation. Some examples and analysis are presented in the third part. A conclusion makes the last part of this paper.

2 GENERAL FRAME OF CO-ROTATIONAL APPROACH

The main concept of CR approach is the abstract frame of Inertial Frame, Co-rotational Frame, it is similar to the inertial and floating frame in the FFRF, and the meanings of both are also basically identical. The core of co-rotational approach is to extract a pure deformation from the overall displacement, which relies on the so called projector. In the next subsection, the projector is derived.

The notation, \( x' \) means the quantity is at the initial configuration, \( x' \) means the quantity is measured in CR frame. \( \bar{x} \) means \( x \) denotes the pure deformation, either the displacement or rotation. \( x \) is in deformed configuration measured in inertia frame.

2.1 Deformation description

Without loss of generality, a triangle element \( e \) experiences a general motion with rigid body and deformation. An inertial frame \((Oxyz)\) is fixed in arbitrary place, and a co-rotational frame \((Cxyz)\) is fixed at the centroid of this element, which are showed in figure 1.
The position of Node $i$, $r_i$ is the sum of the position of the CR $r_e$ and relative position with respect to CR $r_{ei}$. And $r_{ei}$ can be expressed by the initial relative position $r_{ei}^0$ and deformation displacement $\bar{u}_i'$.

$$r_i = r_e + r_{ei}, \quad r_{ei} = R_e (r_{ei}^0 + \bar{u}_i')$$  \hspace{1cm} (1)

where $R_e$ is the rotation matrix of CR frame denoting the orientation of CR frame.

The variation of (1),

$$\delta u_i = \delta u_e - \bar{r}_e \delta \psi_e + R_e \delta \bar{u}_i'$$  \hspace{1cm} (2)

where $\delta R_e = \delta \bar{\psi}_e R_e$ with the consideration of $\delta u_e = \delta r_e$, $\delta r_{ei}^0 = 0$.

In the local coordinate system, or in CR frame, the (2) becomes

$$\delta \bar{u}_i' = \bar{r}_{ei}' \delta \psi_e' + \delta u_i' - \delta u_e'$$  \hspace{1cm} (3)

For nodal rotation, the overall rotation of the Node $i$ expressed by $R_i$, which can be rewritten by the CR frame rotation,

$$R_i = R_e \bar{R}_i$$  \hspace{1cm} (4)

which means the $R_i$ is completed by a compound rotation about follower axes, first rotation by $R_e$ with a axis of fixed, and the axis of second rotation is transported to operating location by the first rotation[9]. Variation of (4)

$$\delta \bar{R}_i' = (\delta \bar{\psi}_e' - \delta \psi_e') \bar{R}_i'$$  \hspace{1cm} (5)

with the consideration of $\delta R_e = R_e \delta \bar{\psi}_e$, $\delta \psi_e = \delta \bar{\psi}_e R_e = R_e \delta \bar{\psi}_e' R_e^T R_i$, where the $\delta \bar{\psi}_e$ is the infinitesimal rotation vector, (5) reveals that the infinitesimal rotation vector be additive in the same frame.

2.2 The projector and transformation matrix
From the description of deformation, the relationship between the overall displacement and deformation displacement can be obtained.

From (3) and (5), $\delta \mathbf{u}_i'$ and $\delta \psi_j'$, should be the function of $\delta \mathbf{u}_j'$ and $\delta \psi_j'$, i.e.

$$
\delta \mathbf{q}_{ij}' = \sum_{j=1}^{3} P_{ij} \delta q_{ij}'
$$

where

$$
\delta q_{ij}' = R_i^T \delta q_{ij}' , \delta \mathbf{q}_{ij}' = \begin{pmatrix} \delta \mathbf{u}_j' \\ \delta \psi_j' \end{pmatrix} , \delta \mathbf{q}_{ij}' = \begin{pmatrix} \delta \mathbf{u}_j' \\ \delta \psi_j' \end{pmatrix}
$$

$$
P_{ij} = \begin{bmatrix}
\frac{\partial \psi_j'}{\partial \mathbf{u}_j'} + U_{ij} \\
\frac{\partial \psi_j'}{\partial \mathbf{u}_j'} \\
-\frac{\partial \psi_j'}{\partial \mathbf{u}_j'} \\
\frac{\partial \psi_j'}{\partial \psi_j'} - \frac{\partial \psi_j'}{\partial \psi_j'}
\end{bmatrix}
$$

$\delta \mathbf{q}_{ij}'$ and $\delta q_{ij}'$ are the deformation displacement and overall displacement, the superscription $\psi$ means the rotation is expressed as infinitesimal rotation vector. Assuming that $U_{ij}$ is a constant matrix that denotes the relation between nodal displacement and the displacement of the origin of CR frame.

Consider the (8), the item $\frac{\partial \psi_j'}{\partial \psi_j'}$ and $\frac{\partial \psi_j'}{\partial \psi_j'}$ mean the orientation relation between CR frame and nodal displacement, which depend on the different choices of CR frame. In general, there are two types of CR frame, one is explicit defined frames that only depends on the geometry of the element, i.e. the x axis of CR frame is aligned to 1-2 side of the element and picks the centroid as origin. The other is implicit defined frames that relys on the deformation of the element, i.e. minimization of the local nodal displacement and zero local spin at centroid of the element, or the polar decomposition.

$P_{ij}$ from (6) extracts the pure deformation displacement from the overall displacement, which means $P_{ij} = 0$ when the element experiences a rigid body motion, $P_{ij} = I$ when the element is under a pure deformation.

$\delta \psi_j'$ from (6) cannot be used to calculate the internal force in CR frame in general, it needs to be transformed to finite rotation variables $\delta \mathbf{u}_j'$ with an assumption that the components of which are small enough, so that the components are uncoupled around the three axies of the CR frame. The transformation is expressed as follows,

$$
\delta \mathbf{q}_i^{\theta} = \mathbf{\hat{H}} \delta \mathbf{q}_{ij}'
$$

where $\mathbf{\hat{H}} = \begin{bmatrix} I \\ T(\mathbf{\hat{u}})' \end{bmatrix}$, $T(\mathbf{\hat{u}})' = \frac{\partial \mathbf{u}_j'}{\partial \psi_j'}$, and which is derived by [8].

The whole transformation is written as

$$
\delta \mathbf{q}_i^{\theta} = \mathbf{\hat{H}} \delta \mathbf{q}_{ij}' = \mathbf{\hat{H}} P \delta \mathbf{q}_{ij}' = \mathbf{\hat{H}} P \mathbf{K}_i \delta \mathbf{q}_{ij}'
$$
In fact $\delta q^\nu$ can be transformed to any type of finite rotation variable, finite rotation vector[10], Euler angles, Cardan angles or the Euler Quaternions[11].

2.3 The tangent stiffness of internal force

The internal energy is done by either overall displacement or pure deformation displacement, $\delta q^\nu \vec{f}^\nu = \delta q^\nu \vec{f}^\nu$, so the similar transformation is as follows,

$$\hat{R}_c P^T \hat{H}^T \vec{f}^\nu = \vec{f}^\nu$$

(11)

where $\vec{f}^\nu$ is the internal force measured in the CR frame, $\vec{f}^\nu$ is the conjugate part of $\delta q^\nu$. Variation of (11) makes the

$$\delta \vec{f}^\nu = \delta \hat{R}_c P^T \hat{H}^T \vec{f}^\nu + \hat{R}_c \delta P^T \hat{H}^T \vec{f}^\nu + \hat{R}_c P^T \delta \hat{H}^T \vec{f}^\nu + \hat{R}_c P^T \hat{H}^T \delta \vec{f}^\nu$$

(12)

where $K_{GR}, K_{GP}, K_{GH}$ are the geometrical stiffness matrix, and $K_M$ is the material stiffness matrix. $\delta \vec{f}^\nu = \vec{K}_M^\nu \delta q^\nu$, $\vec{K}_M^\nu$ is the linear stiffness matrix which can be replaced by any stiffness matrix derived by arbitrary linear element. That's the reason why the most advantage of Co-rotational formulation is reusing element or element independent. (12) is so called consistent tangent stiffness matrix[8].

3 ROTATION FREE ELEMENT

Consider the (12), the item $K_{GR}, K_{GH}$ are including most complicated computation that costs the most of time in calculating the tangent stiffness matrix, and the rotation variables in $K_G$, are the most part of all the item, and because of the rotation variables, the transformation matrix may be singular at integration.

For a shell-like structure, the rotation variables express the curvature of a mid-surface, the curvature can be computed by rotation-free element, which can settle the efficiency and singular problem.

Rotation free element transforms the area integration of control domain into a line integration, and the derivative of the deflection be computed by a set of triangular cells.

**Figure 2.** The control domain of triangle cells.

Figure 2 shows the control domain in shaded triangles with the external normal around the element. The curvature of control domain is computed by the element around the control domain. The curvature of control domain[12,13] is as follows,

$$\kappa_p = \frac{1}{A_p} \sum_{j=1}^{3} \frac{T_j}{2} \left( \nabla N^{(p)} w^{(p)} + \nabla N^{(j)} w^{(j)} \right) = B_p w$$

(13)
From (14), the curvature can be expressed by deflection of triangle element, can (12) can be replaced by

\[
\delta f'' = \delta \hat{\mathbf{r}} \mathbf{P}^T \hat{\mathbf{H}}^T \mathbf{j}'' + \hat{\mathbf{r}} \delta \mathbf{P}^T \hat{\mathbf{H}}^T \mathbf{j}'' + \hat{\mathbf{r}} \mathbf{P}^T \hat{\mathbf{H}}^T \delta \mathbf{j}''
\]

\[
= (k_{GP} + k_{GP}'' + k_{M}) \delta q''
\]

where \( k_{GP}'' \) with no rotation variables, the computation is much simpler than ever and \( \delta \hat{\mathbf{H}}^T \), the transformation of infinitesimal rotation vector is eliminated.

4 NUMERICAL EXAMPLES

4.1 Hemispherical shell subjected to alternating radial forces

A hemispherical shell with an 18 degree circular cutout at its pole. The shell is loaded by alternating radial point force at 90 degree intervals.

The parameters of hemispherical shell, \( E=6.825e7 \), \( v=0.3 \), \( R=10 \), \( h=0.04 \), \( P_{\text{max}}=400 \).

![Figure 3. Hemispherical shell.](image)

4.2 Pullout of an open-ended cylindrical shell

Figure 4 shows a open-ended cylinder being pulled by a pair of radial forces.

The shell parameters, \( E=10.5e6, v=0.3125, R=4.953, L=10.35, h=0.94, P_{\text{max}}=40000 \).
4.3 Deployment of a Solar array

A solar array with 18 solar cell, width and length of one cell are 1 m, depth 0.005, under a couple of force loaded at the first Solar array. A relative large scale problem can be settled by CR approach conveniently.

The solar cell is fixed to each other at the end of cells, there is no any joint or constraint. The parameters of cells are $E=10e6$, $v=0.3125$. The deployment procedure is done by a couple of force, the energy done by the external force completely become internal energy. This example shows the efficiency of the CR formulation, which solve the problem in less time and iteration step compared to geometrical non-linear theory.

5 CONCLUSIONS

The CR formulation concentrates in the geometrical non-linear problem, all types of elastic problem can be settled by CR formulation. The general CR formulation with rotation-free element leads to a simple and efficient formulation rather than that with rotation variables. Lots of examples verified the accuracy and efficiency. For the future research, to promote the performance of the CR formulation, a simplified tangent stiffness matrix should be obtained, and for more intensive computation, an modified program modified by OpenMP(parallel API) and Pardiso(sparse solver) is effective.

REFERENCES


New formulation of the rigid finite element method and its application to modelling of lines and risers

Iwona Adamiec-Wójcik, Łukasz Drąg, Stanisław Wojciech

Faculty of Management and Transport
University of Bielsko-Biała
Willowa 2, 43-309 Bielsko-Biała, Poland
[i.adamiec, ldrag, swojciech]@ath.bielsko.pl

ABSTRACT

The widespread use of lines, cables and flexible risers in offshore applications requires research on the dynamic behavior of such systems. The static and dynamic analysis of slender systems requires large deformations to be taken into consideration. This paper presents a modification of the rigid finite element method which enables modelling of such systems to include bending, torsional and longitudinal flexibility. The method consists in dividing a link into rigid finite elements assuming inertial features connected by massless spring-damping elements; additionally, a new spring element reflecting longitudinal flexibility is added. The formulation of the method presented allows us to take into account hydrodynamic forces and added mass. Correctness of the models and programs developed is proved by comparison of the results with an analytical solution and commercial software. Numerical simulations presented in the paper concern the movement of a vessel with an attached riser without fluid in contact with an obstacle on the seabed.

Keywords: rigid finite element method, slender links, longitudinal flexibility, flexible risers.

1 INTRODUCTION

In problems of dynamics and control in offshore engineering long, slender elements undergoing large base motion often have to be considered. Those elements are subject to forces specific for the marine environment caused by waves and sea currents, and also hydrodynamic resistance and buoyancy forces. In order to discretise the slender elements usually the finite element [1], the finite segment [2,3] and lumped mass [4] methods are used. Review of the methods used for modelling multibody systems with flexible links for last more than thirty years is presented in [5]. However, applicability of methods depends not only on their universality but also on numerical effectiveness, which involves both efficiency of the procedures of integrating the equations of motion and also the degree to which simplifications permissible from the engineer’s point of view are possible; these simplifications determine the number of degrees of freedom of the system considered and thus influence the calculation time. A numerically effective method is also the rigid finite element method (RFEM) [6-8], especially its new formulation for beams, lines and risers presented in the paper. This approach enables shear, torsion or longitudinal stiffness to be eliminated without the necessity of reformulating the mathematical model of the system considered. Slender links are discretized by dividing them into rigid finite elements which assume the mass (inertial) characteristics of the link, and into massless, dimensionless spring-damping elements which assume the flexible characteristics. The method of discretisation of a flexible link into rigid finite elements (rfe) and spring damping elements (sde) is shown in Figure 1. In the classical formulation, a link with length $L$ is divided into $n$ segments with length $\Delta l$ in so-called primary division (Fig. 1.a):

$$\Delta l = \frac{L}{n}$$

where $L$ is the length of the link (part of line, cable, pipe), $n$ is the number of elements. Spring-damping elements (sde), which reflect flexible features, are placed in the middle of the
segments, and in the next stage, called secondary division, rigid finite elements (rfe) are placed between two sdes as well as between the ends of the link and sdes. They are treated as rigid bodies and assume mass features of the link (Fig. 1b).

In the classical formulation of the method local coordinate systems \( \{i\}' \) are placed in the center of mass of the element (Fig. 2a) and the axes of the system coincide with its principal central inertial axes of the element.

The position of the element is described with respect to the inertial coordinate system by six generalized coordinates which are the elements of the following vector

\[
q^c_i = [x_{C_{i,1}}, x_{C_{i,2}}, x_{C_{i,3}}, \varphi_{i,1}, \varphi_{i,2}, \varphi_{i,3}]^T
\]

where \( x_{C_{i,1}}, x_{C_{i,2}}, x_{C_{i,3}} \) are coordinates of the center of mass of element \( i \), \( \varphi_{i,1}, \varphi_{i,2}, \varphi_{i,3} \) are \( Z'Y'X' \) Euler angles defined for element \( i \) in the local coordinate system; these angles are also used in airplane mechanics and are called heading, attitude and bank angles [9].

Mass parameters of rfe and stiffness coefficients of sdes can be calculated according to formulae presented in [7]. Spring-damping elements reflect longitudinal, shear, bending and torsional flexibilities of elements from primary division. The continuity of displacements is not preserved in the classical formulation.

This way of discretisation enables the matrix of transformation from the local coordinate system to the global one to be defined in the form:

\[
r_i^c = B_i^c(q^c_i)r_i'
\]

where \( r_i' \) are coordinates of rfe \( i \) in the local system of reference, \( r_i \) are coordinates of rfe \( i \) in the global system of reference, \( B_i^c \) is the homogenous transformation matrix [7]. Mass matrices of rfe are uncoupled and the mass matrix of the system of rfe \( 0, ..., n \) takes the form:
\[ M = \text{diag} \{M_0, M_1, \ldots, M_n\} \]  

The classical formulation of the method has also its modification which enables longitudinal and shear stiffness to be eliminated (Fig.2b). The motion of each rfe is defined with respect to its predecessor using angles \( \varphi_{i,1}, \varphi_{i,2}, \varphi_{i,3} \) implied as in the classical approach. The vector of generalised coordinates of rfe \( i \) is as follows:

\[ q_i^m = [\varphi_{i,1}, \varphi_{i,2}, \varphi_{i,3}]^T \]  

Modification of the method ensures the continuity of displacements and eliminates high frequencies by omitting longitudinal and shear stiffness. The disadvantage lies in the way of defining transformation of coordinates:

\[ r_i = B_i^m \begin{pmatrix} q_0^m, \ldots, q_{i-1}^m, q_i^m \end{pmatrix} r_i' \]  

which leads to the full mass matrix. Thus the block-diagonal form characteristic for the classical approach is lost.

2 FORMULATION OF THE METHOD PROPOSED

In the new approach presented in the paper the primary and secondary divisions are carried out as in the classical method (Fig. 1), while the sdes \( \otimes \) reflect only bending and torsional flexibility of elements. They do not reflect shearing and longitudinal flexibility. Subsequently an additional division is performed and as a result an additional spring-damping element \( \Box \) reflecting longitudinal flexibility of the element from secondary division is placed in the middle of each rfe (Fig. 3a).

![Figure 3](image)

**Figure 3.** New concept of the RFEM a) discretization b) generalized coordinates

The local coordinate systems are assigned at the preceding sde \( \otimes \) (Fig.3b) like in the modification of the method mentioned in the previous section. The motion of the elements, which because of the sde \( \otimes \) with longitudinal flexibility should not be called rigid, is described in space by coordinates \( (x_{i,1}, x_{i,2}, x_{i,3}) \) of the beginning of the element (point \( A_i \)), Euler angles \( Z'Y'X' \) \( (\varphi_{i,3}, \varphi_{i,2}, \varphi_{i,1}) \) and displacement \( \Delta_i \) as in Figure 3b.
The motion of part 1 of the element is described by the coordinates of the following vector:

\[ \mathbf{q}_1^{(1)} = [x_{1}, x_{2}, x_{3}, \phi_{1}, \phi_{2}, \phi_{3}]^T = [q_{1,1}^{(1)}, q_{1,2}^{(1)}, q_{1,3}^{(1)}, q_{1,4}^{(1)}, q_{1,5}^{(1)}, q_{1,6}^{(1)}]^T \]  \hspace{1cm} (7.1)

and of part 2 by the coordinates of the vector:

\[ \mathbf{q}_1^{(2)} = [x_{1}, x_{2}, x_{3}, \phi_{1,2}, \phi_{2,3}, \Delta_i]^T = [q_{1,1}^{(2)}, q_{1,2}^{(2)}, q_{1,3}^{(2)}, q_{1,4}^{(2)}, q_{1,5}^{(2)}, q_{1,6}^{(2)}]^T \]  \hspace{1cm} (7.2)

which is also the vector of generalized coordinates of the element.

Transformations of coordinates from the local frame \{i\}' to the global frame of reference \{\}\ are performed using homogenous transformations according to the following formulae:

\[ \mathbf{r}_i^{(j)} = \mathbf{B}_i^{(j)} \mathbf{r}_i^{(j)} \]  \hspace{1cm} (8)

where \( \mathbf{r}_i^{(j)} = [x_{i,j}^{(j)} x_{i,2}^{(j)} x_{i,3}^{(j)} 1]^T \) – coordinates of the point belonging to part \( j = \{1, 2\} \) of the element in local frame \{i\}', \( \mathbf{r}_i^{(j)} = [x_{i,j}^{(j)} x_{i,2}^{(j)} x_{i,3}^{(j)} 1]^T \) are coordinates of the point belonging to part \( j \) of the element in global frame \{\}\, \( \mathbf{B}_i^{(1)} = \mathbf{B}_i^{(0)} \mathbf{B}_i^{(1)} \), \( \mathbf{B}_i^{(2)} = \mathbf{B}_i^{(0)} \mathbf{B}_i^{(2)} \mathbf{B}_i^{(\alpha)} \), \( \mathbf{B}_i^{(\alpha)} \) is the rotation matrix,

\[ \mathbf{B}_i^{(1)} = \begin{bmatrix} 1 & 0 & 0 & x_{i,1} \\ 0 & 1 & 0 & x_{i,2} \\ 0 & 0 & 1 & x_{i,3} \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{B}_i^{(2)} = \begin{bmatrix} c_{i,2} & 0 & s_{i,2} & 0 \\ -s_{i,2} & c_{i,2} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{B}_i^{(\alpha)} = \begin{bmatrix} 1 & 0 & \Delta_i \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \]

\( \mathbf{c}_{i,p} = \cos \varphi_{i,p}, \mathbf{s}_{i,p} = \sin \varphi_{i,p} \) \( p = 1, 2, 3 \).

The equations of motion are derived from the Lagrange equations and described in detail in [10].

Because it is assumed that the motion of elements 0 to \( n \) is described by independent coordinates of vectors \( \mathbf{q}_i \), constraint equations and their reactions have to be introduced in the system. Constraint reactions for element \( i \) of a slender link are shown in Figure 4.

![Figure 4. Element i with constraint reactions](image)

Coordinates of point \( A_{i+1} \) depend on generalised coordinates of element \( i \) as follows:

\[ \mathbf{r}_{\Delta i} = \mathbf{r}_i + \mathbf{R}_i \mathbf{r}_i^{(\alpha)} \]  \hspace{1cm} (9)

where \( \mathbf{r}_i = [x_{i,1}, x_{i,2}, x_{i,3}]^T \), \( \mathbf{r}_{\Delta i} = [x_{i+1,1}, x_{i+1,2}, x_{i+1,3}]^T \), \( \mathbf{R}_i = \mathbf{R}_i^{(3)} \mathbf{R}_i^{(2)} \mathbf{R}_i^{(1)} \), \( \mathbf{R}_i^{(j)} \) are rotation matrices obtained from \( \mathbf{B}_i^{(j)} \) by extracting first three columns and rows, \( l_i \) is the length of rfe \( i \).
Generalised forces resulting from reactions in $A_i$ and $A_{r+1}$ can be written in the form:

$$
Q_i = \left( \frac{\partial r_i}{\partial q_i} \right)^T F_i - \left( \frac{\partial r_{r+1}}{\partial q_i} \right)^T F_{r+1}
$$

(10)

where $F_i = [F_{i,1} \ F_{i,2} \ F_{i,3}]^T$, $F_{r+1} = [F_{r+1,1} \ F_{r+1,2} \ F_{r+1,3}]^T$, $Q_i = [Q_{i,1} \ \ldots \ Q_{i,r} \ ]^T$.

Thus, we obtain the following:

$$
Q_i = \vec{D} F_i + D_i F_{r+1}
$$

(11)

where $\vec{D} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$; $D_i = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \\ (l_i + \Delta_i) s_{i,2} c_{i,3} & (l_i + \Delta_i) s_{i,2} s_{i,3} & (l_i + \Delta_i) c_{i,2} \\ (l_i + \Delta_i) c_{i,2} s_{i,3} & - (l_i + \Delta_i) c_{i,2} c_{i,3} & 0 \\ - c_{i,2} c_{i,3} & - c_{i,2} s_{i,3} & s_{i,2} \end{bmatrix}$.

The models of links presented are used in the case when motion of point $A_0$ is known, which means that we know the vector:

$$
r_0 = r_0(t)
$$

(12)

In practice vector $r_0$ can describe motion of any point of the considered structure, for example caused by sea waves or a vessel movement.

The equations of motion of elements from 0 to $n$ of the slender link can be written in the form:

$$
A_i \ddot{q}_i - \vec{D} F_i - D_i F_{r+1} = Q_i \left( t, q_{i-1}, q_i, \dot{q}_{i-1}, \dot{q}_i, \ddot{q}_{i-1}, \ddot{q}_i, \dddot{q}_{i-1}, \dddot{q}_i \right)
$$

(13)

Forces $Q_i$ include moments transferred by spring-damping elements $\otimes$.

### 3 EQUATIONS OF MOTION

The sea environment impact forces, for examined flexible elements of large length and small cross-section, can be introduced using Morison equations [11]. If forces caused by flow inside the element are disregarded, then consideration of water environment impact requires taking into account buoyant force, viscous resistance forces and forces of inertia. Consideration of those forces leads to formulation of generalised forces caused by water environment:

$$
Q_i^M = \vec{A}_i (q_i, \dot{q}_i, \ddot{q}_i) + \vec{h}_i (q_i, \dot{q}_i)
$$

(14)

The first component in Eq(14) is connected with the mass of added water. Having taken into account the above forces the equations of motion of ref 0,...,n can be written as follows:

$$
M_i \ddot{q}_i - \vec{D} F_i - D_i F_{r+1} = \vec{Q}_i
$$

(15)

where $M_i = A_i - \vec{A}_i$, $\vec{Q}_i = Q_i + \vec{h}_i$. 

404
If we introduce the following vectors [10]:

\[ \mathbf{q} = \begin{bmatrix} \mathbf{q}_0^T & \cdots & \mathbf{q}_i^T & \cdots & \mathbf{q}_n^T \end{bmatrix}^T \quad \text{generalised coordinates of the link,} \quad (16.1) \]

\[ \mathbf{F} = \begin{bmatrix} \mathbf{F}_0^T & \cdots & \mathbf{F}_i^T & \cdots & \mathbf{F}_n^T \end{bmatrix}^T \quad \text{reaction forces,} \quad (16.2) \]

\[ \mathbf{Q} = \begin{bmatrix} \mathbf{Q}_0^T & \cdots & \mathbf{Q}_i^T & \cdots & \mathbf{Q}_n^T \end{bmatrix}^T \quad \text{generalised forces,} \quad (16.3) \]

\[ \mathbf{G} = \begin{bmatrix} \mathbf{G}_0^T & \cdots & \mathbf{G}_i^T & \cdots & \mathbf{G}_n^T \end{bmatrix}^T \quad \text{right sides of constraint equations,} \quad (16.4) \]

and matrices:
- mass matrix:

\[ \mathbf{A} = \text{diag}(A_0, \ldots, A_i, \ldots, A_n) \quad (17.1) \]

- matrix of coefficients of constraint reactions:

\[ \mathbf{D} = \begin{bmatrix} \mathbf{D} & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & \mathbf{D} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \end{bmatrix} \quad (17.2) \]

the equations of motion and the constraint equations can be written as follows:

\[ \mathbf{A}\ddot{\mathbf{q}} - \mathbf{D}\dot{\mathbf{F}} = \mathbf{Q} \quad (18.1) \]

\[ \mathbf{D}^\top \dot{\mathbf{q}} = \mathbf{G} \quad (18.2) \]

Due to acceleration form of the constraint equations a stabilization algorithm was used.

4 NUMERICAL SIMULATIONS

4.1 Verification of the model

This section validates the model by comparing the results obtained with those presented by Raman-Nair and Baddour in [4]. The authors consider an elastic catenary without bending stiffness and use the closed solution presented by Irvine [12]. Figure 5 presents the system considered.

The results presented below are obtained for the following data: length of the unstretched line \( L = 100 \text{m} \), area of the cross-section \( A = \pi (0.035)^2/4 \text{m}^2 \), Young’s modulus \( E = 10^{11} \text{N/m}^2 \), density \( \rho = 2 \times 10^3 \text{kg/m}^3 \), horizontal force \( H = 1000 \text{N} \), vertical force \( V = 2000 \text{N} \).
Figure 6 shows the comparison of curves $y(x)$ obtained for the analytical solution and for the RFEM assuming $n=10$. It can be seen that the broken line obtained by the method accurately represents the course of the catenary.

![Figure 6. Analytical solution and RFEM for $n=10$](image)

### 4.2 Validation of the model

Jensen and co-authors in [13] derive a model of a pipe submerged in water using nonlinear elastic beam equations, which are then discretized by the FEM. They consider an installation of a steel pipe by means of the J-lay method. A natural catenary equation is used for validation of the model of the pipe without bending stiffness, while for the validation of the model when bending stiffness is included the authors present results obtained from RIFLEX. For the static case horizontal force $H$ with three different values is applied at the end of the pipe as in Figure 7.

![Figure 7. J-lay installation of a pipe with horizontal force](image)

The parameters of the system used for the analysis are presented in Table 1.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Parameter</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L$</td>
<td>length</td>
<td>1500 m</td>
<td>m</td>
</tr>
<tr>
<td>$D_{out}$</td>
<td>outer diameter</td>
<td>0.762 m</td>
<td>m</td>
</tr>
<tr>
<td>$D_{in}$</td>
<td>inner diameter</td>
<td>0.696 m</td>
<td>m</td>
</tr>
<tr>
<td>$E$</td>
<td>Young’s modulus</td>
<td>$2.07\times10^4$ N/m$^2$</td>
<td></td>
</tr>
<tr>
<td>$\rho_r$</td>
<td>riser density</td>
<td>7850 kg/m$^3$</td>
<td></td>
</tr>
<tr>
<td>$\rho_w$</td>
<td>water density</td>
<td>1025 kg/m$^3$</td>
<td></td>
</tr>
<tr>
<td>$C_M$</td>
<td>coefficient</td>
<td>2</td>
<td>-</td>
</tr>
</tbody>
</table>
Table 2 presents values of hang-off angle $\beta$ and lay-back distance $l_h$ for cases with and without bending stiffness presented by Jensen et al. (2010), and those obtained using the RFEM with $n = 300$ elements. Seabed interaction is described by the same formulae.

<table>
<thead>
<tr>
<th>Method</th>
<th>$H=200\text{kN}$</th>
<th>$H=400\text{kN}$</th>
<th>$H=800\text{kN}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\beta$</td>
<td>$l_h$</td>
<td>$\beta$</td>
</tr>
<tr>
<td>without</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bending</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RFEM</td>
<td>81.21</td>
<td>412.26</td>
<td>74.63</td>
</tr>
<tr>
<td>FEM</td>
<td>81.22</td>
<td>412.25</td>
<td>74.64</td>
</tr>
<tr>
<td>Catenary</td>
<td>81.22</td>
<td>415.25</td>
<td>74.64</td>
</tr>
<tr>
<td>with</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bending</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RFEM</td>
<td>80.98</td>
<td>467.23</td>
<td>74.31</td>
</tr>
<tr>
<td>FEM</td>
<td>80.97</td>
<td>467.92</td>
<td>74.30</td>
</tr>
<tr>
<td>RIFLEX</td>
<td>81.00</td>
<td>477.00</td>
<td>74.40</td>
</tr>
</tbody>
</table>

Analysis of results from Table 2 proves that the Rigid Finite Element Method for $n=300$ gives almost the same results as the exact beam elements with the Finite Element Method when the number of elements equals 740. This is due to the fact that linearization is not carried out in the formulation of the RFEM presented.

4.3 Transport of the riser over an obstacle

In the case of bad weather conditions or failure, risers are disconnected from the wellhead and moved. When the conditions allow, the riser is connected to the wellhead again. It is possible that in such situations the riser connected to the holding vessel will be moved. Xu and Wang [14] present an interesting model and results of analysis for such a case. In this paper we consider a situation in which the movement of the bottom end of the pipe can be restrained by the roughness of the seabed (Fig. 8).

![Figure 8. Scheme of the movement of the riser](image)

Velocity of the vessel $\dot{x}_0$ in $x$ direction changes as in Figure 9, while coordinate $y_0$ is defined by the relation:

$$y_0 = a_y \sin \left( \frac{2\pi}{T_y} t + \frac{\pi}{2} \right)$$  \hspace{1cm} (19)

where $a_y$, $T_y$ are constant.
Simulation results are presented for various cases, which we denote as:

P1: horizontal movement of the vessel — lack of the obstacle ($H_0 = 0$);
P2: horizontal movement of the vessel — with the obstacle ($H_0 = 8$ m);
P3: vertical and horizontal movement of the vessel with amplitude of $0.25$ m — with the obstacle ($H_0 = 8$ m).

Calculations have been carried for a riser discretised into $n=150$ elements. The equations of motion have been integrated by means of the classical explicit Runge-Kutta method of the fourth order with a constant integration step $h=0.001$.

Figure 10 presents trajectories of point $E$ of the riser for cases P1, P2 and P3 respectively.

## 5 CONCLUSIONS

The modification of the rigid finite element method presented in the paper is more numerically efficient than the previous formulations of the method. Elimination of shear stiffness does not exclude the possibilities of calculation of shear forces since the model enables us to calculate the reaction forces in connections between rigid elements. Such an approach eliminates vibrations of high frequencies and thus the integration step can be considerably larger. Additional elimination of longitudinal or torsional flexibility further shortens the time of calculations.
REFERENCES


Nonlinear Vibration of Cantilever under the Action of Lateral Harmonic Excitations and Axial Load

Guo-Kang Er, Haien Du

Department of Civil and Environmental Engineering
University of Macau, Macau SAR, China
mb25488@umac.mo

ABSTRACT

In this paper, a uniform cantilever beam under the action of the lateral harmonic load on the tip and axial self-weight has been studied. In order to find the effects of high-order nonlinear terms on the response, such as the third and fifth-order nonlinear terms, the variational approach based on extended Hamilton principle is employed to derive the equations of motion and boundary conditions governing the planar nonlinear vibrations of the isotropic and inextensible Euler-Bernoulli beams. Three second-order partial differential equations containing third and fifth-order nonlinear terms are obtained. With Galerkin method, the nonlinear dynamical system in time domain is formulated. Then Runge-Kutta method is adopted to analyse the responses of the system. The nonlinear behavior of the cantilever beam has been examined. The nonlinearity of the cantilever beam under the lateral harmonic load on the tip and the axial self weight is investigated with different frequencies of harmonic excitations and axial forces. Some hardening and softening behaviors have been observed and discussed through the numerical analysis.

Keywords: Nonlinear vibration, Cantilever beam, Hamilton principle, Galerkin method, Runge-Kutta method.

1 INTRODUCTION

Cantilever beam is one of mostly used engineering structure. It can be found in various structural applications. Structures like high-rise buildings, helicopter rotor blades, and subsystems of some complex structures. Many works about the vibration of linear and nonlinear cantilever beam containing up to third-order nonlinear terms have been done. In order to model the systems accurately, one needs to incorporate nonlinearity in the model. Especially when the system undergoes large motions, the geometric nonlinearity and inertial nonlinearity are activated and their effect cannot be ignored. Under such circumstance, the effects of higher-order nonlinear terms need being taken into account. Generally, there were two categories in the analytical investigations of the responses of cantilever beams. In the first category, the influence of nonlinearity is either neglected or partially considered. Silva and Glynn used the equations they derived to investigate the flexural-flexural responses of near-square cantilever beams subject to primary resonances (1978a, b) [1, 2]. They found that the response amplitude-frequency curves show much different behavior at the first natural frequency from those at higher natural frequencies. Furthermore, the influence of the nonlinear curvature terms on the response diminishes for higher modes. Nayfeh and Pai (1989) [3] used the equations of Silva and Glynn (1978b) [2] to investigate the nonlinear nonplanar responses of cantilever beams subject to parametric excitations in the presence of a one-to-one internal resonance involving two flexural modes. Pai and Nayfeh (1990a) [4] also used the equations of Silva and Glynn (1978b) [2] to investigate the nonplanar oscillations of square and rectangular cantilever beams subject to lateral base excitations. They found that the geometric nonlinearity dominates the inertia nonlinearity for the low-frequency modes, whereas the inertia nonlinearity dominates the geometric nonlinearity for the high-frequency modes. The geometric nonlinearity produces a hardening effect and the inertia nonlinearity produces a softening effect. Crespo da Silva, Zaretzky, and Hodges (1991) [5], in a work related to that of Dowell, Traybar, and Hodges (1977) [6], analytically investigated the accuracy of approximate solutions for the free
nonplanar response of cantilever beam-mass systems. They found excellent agreement with the results obtained by numerically integrating the exact equations, even for large tip loads. Shyu, Mook, and Plaut (1993a, b, and c) [7, 8 and 9] investigated the nonlinear response of square cantilever beams subject to transverse harmonic and non-stationary excitations. In addition to the cubic geometric and inertia nonlinearities, they accounted for the effect of the static deflection due to the beam's weight, which introduced quadratic nonlinearities in the governing equations and found that whirling motions are possible and in some cases they are the only stable motions. Furthermore, they found that increasing the damping can reduce the amplitudes of the whirling motions. Crespo da Silva and Zaretzky (1994a) [10] investigated the nonlinear flexural-flexural-torsional responses of cantilever beams. Lee, Lee, and Chang (1997) [11] used the equations of Crespo da Silva and Zaretzky (1994a) [10] to investigate the nonlinear flexural-flexural-torsional responses of cantilever rectangular beams subject to harmonic torsional base excitations in the presence of a one-to-one internal resonance. They found that two mode oscillations occur as a result of the losing stability in torsion motion. Afafat and Nayfeh (2001) [12] used the equations derived by Crespo da Silva and Glynn (1978b) [2] and multiple scales method to investigate the effects of the nonlinear boundaries. Eftekhari, Mahzoon and Ziaei-rad (2012) [13] used multiple scales method to carry out a comparative study between a beam with and without a tip mass and found that when the excitation frequency or amplitude of excitation force is slowly changed, the presence of tip mass altering the stability of solutions. In this paper, a uniform cantilever beam under the action of the lateral harmonic load on the tip and axial self-weight has been studied. In order to find the effects of high-order nonlinear terms on the response, such as the third and fifth-order nonlinear terms, the variational approach based on extended Hamilton principle is employed to derive the equations of motion and boundary conditions governing the planar nonlinear vibrations of the isotropic and inextensible Euler-Bernoulli beams. Three second-order partial differential equations containing third and fifth-order nonlinear terms are obtained. With Galerkin method, the nonlinear dynamical system in time domain is formulated. Then Runge-Kutta method is adopted to analyse the responses of the system. The nonlinear behavior of the cantilever beam has been examined. The nonlinearity of the cantilever beam under the lateral harmonic load on the tip and the axial self weight is investigated with different frequencies of harmonic excitations and axial forces. Some hardening and softening behaviors have been observed and discussed through the numerical analysis.

2 PROBLEM DESCRIPTION

Large deformation or large displacement of a structure does not neccessarily mean the existence of large strains. Structures like cantilever beams can undergo large deformations but small strains. For flexible cantilever, the large deformation can lead to obvious nonlinear behavior. Since the large deformation gives rise to geometric nonlinearities due to nonlinear curvature and neutral-axis stretching or shortening, it leads to nonlinear strain-displacement relations. We consider the beam to be analyzed as a nonlinear elastic structure. The Euler-Bernoulli beam theory (Shames and Dym, 1985) [14] is adopted to model the beam, by neglecting the effects of warping and shear deformation. To simplify the expressions, the influence of Poisson effect and torsion in the axial direction are also neglected. In the absence of warping, the differential beam element can be considered as a rigid body, whose planar motion is then completely described by two translational displacements and one rotational displacement as shown in the following.

2.1 Equations of Motion

A uniform and initially straight beam with length \( L \) and mass per unit length \( m \) is fixed at one end and a harmonic load is applied at the another as shown in Figure 1. The self-weight of this cantilever beam is considered through out the whole solution procedure. In Figure 1, \( (x, y, z) \) denotes the inertial coordinate system with orthogonal unit vector \( (e_x, e_y, e_z) \) being \( e_x = (1, 0, 0) \), \( e_y = (0, 1, 0) \) and \( e_z = (0, 0, 1) \), and \( (\xi, \eta, z) \) denotes a local curvilinear
coordinate system with mutually orthogonal unit vectors \( \{e_x, e_y, e_z\} \) at arclength \( s \) in the deformed position. Since the beam has uniform cross section and material properties, its mass and area centroids are identical and the axes \( (\xi, \eta) \) are taken to be the principle axes of the beam’s cross section at any position \( s \). Moreover, the \( x \) and \( \xi \) axes represent the neutral axis of the beam before and after deformations, respectively. With the above assumptions, each cross section of the beam undergoes an elastic displacement of its centroid \( C \) and a rotation about \( z \) axis at point \( C \). \( u(s,t) \) and \( v(s,t) \) in Figure 1 represent the displacement components of the centroid \( C \) with respect to the inertial axes \( x \) and \( y \) at any arclength \( s \) and time \( t \), respectively. The rotation from the undeformed to the deformed position about centroid \( C \) is described by one counterclockwise Euler angle rotation and denoted as \( \psi(s,t) \) as shown in Figure 2.

![Figure 1. A scheme of a cantilever beam with planar flexural motion.](image)

The rotation \( \psi \) about the out-of-plane axis \( z \) in Figure 1 transfers the unit vectors \( \{e_x, e_y\} \) to \( \{e_\xi, e_\eta\} \). The transformation from \( \{e_x, e_y\} \) to \( \{e_\xi, e_\eta\} \) can be expressed as:

\[
[T_\psi] = \begin{bmatrix}
\cos \psi & \sin \psi \\
\sin \psi & -\cos \psi
\end{bmatrix}
\]

so the unit vectors are related by the following equation.

\[
\begin{bmatrix}
e_\xi \\
e_\eta
\end{bmatrix} = [T_\psi] \begin{bmatrix}
e_x \\
e_y
\end{bmatrix}
\]

From Figure 2, it is seen that the absolute angular velocity \( \omega(s,t) \) of the principal axis \( z \) can be expressed as

\[
\omega(s,t) = \psi e_z = \omega_z e_z
\]
where the overdot stands for $\partial / \partial t$. According to the Kirchhoff's kinetic analogue (Love, 1944) [15], the equations of a thin rod subjected only to end forces has the same form as those of a rigid body oscillating about a fixed point. The curvature component due to beam bending can be obtained by the definition of curvature [15].

$$\rho_z \equiv \frac{\partial e_z}{\partial s} \cdot e_\eta$$  \hspace{1cm} (4)

where the dot denotes the inner product of two vectors. Then the Kirchhoff's kinetic analogue can be used to obtain the expressions for the components of the curvature vector $\rho(s,t)$ by simply replacing the time derivatives with the spatial derivatives in the angular velocity expression. Thus, from Equation (3), we have

$$\rho(s,t) = \psi' e_z = \rho_s e_z$$  \hspace{1cm} (5)

where the prime stands for $\partial / \partial s$.

There are totally three independent variables, $(u, v)$ and $\psi$. It is well-known that in the absence of large axial forces, the members with fixed-free ends can be considered as inextensional members. To simplify the analysis, it is assumed that the neutral axis of the beam is inextensional in addition to no warping or shear deformation assumptions. Now we consider the deformation of an element $CD$ of the beam's neutral axis, which is of length $ds$ and located at distance $s$ from the origin $O$ of the $(x, y, z)$ system as shown in Figure 3.
After deformation, $CD$ move to new position $C'D'$. In plane, we denote the displacement components of $C$ and $D$ by $(u, v)$ and $(u + du, v + dv)$, respectively. From Figure 3 and with the definition of curvature, the strain $e$ at point $C$ can be expressed as:

$$e = \frac{ds^* - ds}{ds} = \sqrt{(ds + du)^2 + dv^2} - ds = \sqrt{(1+u')^2 + v'^2} - 1$$

(6)

To satisfy the inextensionality assumption, we need to set the strain on the neutral axis to be zero, i.e. $e = 0$. Thus, the inextensionality assumption turns to be

$$(1+u')^2 + v'^2 = 1$$

(7)

Because no shear deformation is considered, the rotation of the cross sections is only owing to bending. Therefore, it is seen from Figure 3 that the angle $\psi(s, t)$ is related to the spatial derivatives of $u(s, t)$ and $v(s, t)$ as

$$\tan \psi = v' / (1+u')$$

(8)

We note that no torsional motions are taken into account here, so the shear strains are zeros. The Euler-Bernoulli beam assumptions of no-transverse-shear and no strains in the plane of the cross section result in the corresponding strain components being equal to zero. For a slender beam undergoing moderate rotations, the coordinates of any point in $\eta, \zeta$ coordinate system are $(\eta, \zeta)$ in the deformed system. Because no shear strains are considered and the curvature component $\rho_z$ is small, the strain components at any point on the cross section are given as

$$\epsilon_{11} = -\eta \rho_z , \epsilon_{12} = \epsilon_{13} = \epsilon_{21} = \epsilon_{22} = \epsilon_{23} = \epsilon_{31} = \epsilon_{32} = \epsilon_{33} = 0 .$$

In order to take into account the constraint of inextensionality and to obtain the differential equations of motion in terms of displacement $v(s, t)$, the Lagrangian $\mathcal{L}$ through a Lagrange multiplier $\lambda$ is introduced in the following.

Let $\mathcal{L}$ denote the lagrangian of motion given as

$$\mathcal{L} = T - V = \int_0^\ell \ell ds$$

(9)

where $T$ is the kinetic energy, $V$ is the potential energy, and $\ell$ is the specific (i.e., per unit length) Lagrangian. The kinetic energy of the beam consists of two parts: translational energy and rotational energy. The translational kinetic energy is given by

$$T_t = \frac{1}{2} m \int_0^\ell \left( u^2 + v^2 \right) ds$$

(10)

and the rotational kinetic energy is given by

$$T_r = \frac{1}{2} \int_0^\ell J_z \omega_z^2 ds$$

(11)

where $J_z$ is the mass moment of inertia about $z$ axis per unit length of the beam and it is defined as
\[ J_z = \iint_A \rho \eta^2 d\eta dz \]  

(12)

in which \( \rho \) denotes the mass density of the beam and \( A \) denotes the area of the cross section of the beam located at a distance \( s \) from the origin of the \((x, y, z)\) system. As the beam is uniform, \( J_z \) is a constant. The kinetic energy can now be written as

\[ T = \frac{1}{2} \int_0^L \left[ m (\dot{u}^2 + \dot{v}^2) + J_z \alpha_z^2 \right] ds \]  

(13)

The potential energy \( V \) can be determined from the corresponding strain energy \( U \) which for the beam considered here is given by

\[ U = \frac{1}{2} \int_0^L \left[ \iint_A (\sigma_{11}) d\eta dz \right] ds \]  

(14)

where the \( \sigma_{11} \) denotes the normal stress in \( \xi \) direction. In Equation (14), it is assumed that the beam is an elastic structure with a linear stress-strain relationship and \( \varepsilon_{22} = \varepsilon_{33} = \gamma_{12} = \gamma_{13} = \gamma_{23} = 0 \). With Hooke’s Law and neglecting Poisson’s effect (to simplify the expressions of the stress components), it gives \( \sigma_{11} \approx E\epsilon_{11} \), where \( E \) is Young’s modulus of the beam material. Substituting this relation and the expressions of strain component into Equation (14), we obtain

\[ V = U = \frac{1}{2} \int_0^L \left[ \iint_A E (-\eta \rho_z)^2 d\eta dz \right] ds \]  

(15)

Equation (15) can be written as

\[ V = \frac{1}{2} \int_0^L D_z \rho_z^2 ds \]  

(16)

where \( D_z = E \iint_A \eta^2 d\eta dz \) is the bending stiffness of the beam.

The Lagrange multiplier \( \lambda(s,t) \) is used to enforce the inextensionality constraint. With Equation (13) and (16) and the inextensionality constraint given by Equation (7), we write the Lagrangian \( \ell \) as follows.

\[ \ell = \frac{1}{2} m (\ddot{u}^2 + \ddot{v}^2) + \frac{1}{2} (J_z \alpha_z^2) - \frac{1}{2} (D_z \rho_z^2) + \frac{1}{2} \lambda \left[ 1 - (1 + \dot{u}^2 - \dot{v}^2) \right] \]  

(17)

The expressions for \( \omega_z \) and \( \rho_z \) within the above equation are given by Equations (3) and (4).

Hamilton's Principle (Meirovitch, 1967) [16] states that, of all the varied paths satisfying the prescribed initial and final configurations, the actual (or true) path extremizes the functional \( I = \int_{t_1}^{t_2} \ell dt \), where \( t_1 \) and \( t_2 \) denote the initial and final time instants. By including the work done by non-conservative forces within the integrand, the extended form of Hamilton principle can be obtained. Using the variation of the functional \( I \) and the fact that the variation and integral operators commute, we can write for the actual path
\[ \delta I = \int_0^L (\delta L + \delta W_{nc}) \, dt = 0 \quad (18) \]

where \( W_{nc} \) denotes the work done by non-conservative forces, such as damping, external forces and moments. This condition of stationarity leads to all of the equations of motion and boundary conditions. Using the generalized forces along the \( x \) and \( y \) axes denoted by \( Q_u \) and \( Q_v \), respectively, and the corresponding damping coefficients denoted by \( c_u \) and \( c_v \), the expression for \( \delta W_{nc} \) can be written as

\[
\delta W_{nc} = \int_0^L \left[ (Q_u - c_u \dot{u}) \delta u + (Q_v - c_v \dot{v}) \delta v \right] \, ds \\
= \int_0^L \left( Q_u \delta u + Q_v \delta v \right) \, ds \quad (19)
\]

Substituting Equations (9), (17) and (19) into Equation (18), we obtain

\[
\delta I = \int_0^L \int_0^L \left( \delta \ell + Q_u' \delta u + Q_v' \delta v \right) \, ds \, dt = 0 \quad (20)
\]

The specific Lagrangian \( \ell \) is a function of \( x_i, (i = 1, 2, \ldots, 5) \) where \( \vec{x} = \{u, \dot{u}, \dot{v}, \psi, \lambda \}^T \). Therefore,

\[
\delta \ell = \sum_{i=1}^{5} \frac{\partial \ell}{\partial x_i} \delta x_i \quad (21)
\]

However there are only two independent variables, namely, \( u \) and \( v \). Variations of the dependent variable \( \psi \) can be obtained using Equation (8) and is given by

\[
\delta \psi = \frac{\partial \psi}{\partial u} \delta u' + \frac{\partial \psi}{\partial v} \delta v' = \frac{-v' \delta u' + (1+u') \delta v'}{(1+u')^2 + v'^2} \quad (22)
\]

The variation and derivative operators commute. Thus, the variations \( \delta \psi' \) and \( \delta \psi'' \) can be written as \( \frac{\partial}{\partial t}(\delta \psi) \) and \( \frac{\partial}{\partial s}(\delta \psi) \), respectively.

Substituting Equation (22) into Equation (21), then, substituting the result in turn into Equation (20), and after performing integrations by parts in Equation (20), we obtain

\[
\int_0^L \left[ \int_0^L \left( -m \dddot{u} + G_u' \right) \delta u + \int_0^L \left( -m \dddot{v} + G_v' \right) \delta v \right] \, ds \, dt = 0 \quad (23)
\]

where \( G_u \) and \( G_v \) represent the operators and given by

\[
G_u = \frac{\partial^2 \ell}{\partial t \partial \psi} \frac{\partial \psi}{\partial u'} + \frac{\partial^2 \ell}{\partial s \partial \psi'} \frac{\partial \psi}{\partial u'} + \lambda (1+u') \quad (24)
\]

\[
G_v = \frac{\partial^2 \ell}{\partial t \partial \psi} \frac{\partial \psi}{\partial v'} + \frac{\partial^2 \ell}{\partial s \partial \psi'} \frac{\partial \psi}{\partial v'} + \lambda v' \quad (25)
\]

Equation (23) is valid for any arbitrary \( \delta u \) and \( \delta v \), which means that the individual integrands are equal to zero. Therefore,
The above equations represent the equations of motion and boundary conditions of a planar flexural beam which are derived using Lagrange's equation.

For the equations of motion formulated above, the exact analytical solution cannot be obtained since they are nonlinear differential equations. The expression of the equations of motions can be simplified by expanding the nonlinear independent variables into polynomials. Then solve the equations numerically. Using the Taylor series expansion and keep nonlinear terms up to fifth-order, we obtain

\[ u' = \sqrt{1 - v'^2} - 1 = -\frac{1}{2} v'^2 - \frac{1}{8} v'^4 + \ldots \]  

\[ \psi = \tan^{-1} \frac{v'}{1 + u'} = \tan^{-1} \left[ \left(1 - v'^2\right)^{-1/2}\right] = v' \left(1 + \frac{1}{6} v'^2 + \frac{3}{40} v'^4\right) + \ldots \]  

The fifth-order expansion of the component in the bending curvature \( \rho \) can be obtained by substituting Equation (30) in Equation (4), and it gives \( \rho_{u} = \psi'' = (1 + \frac{1}{2} v'^2 + \frac{3}{8} v'^4) v'' \).

For slender beams, we assume that the influence of the mass moment of inertia of the beam on beam motion is negligible comparing to the translational inertial. Substituting Equations (29) and (30) into Equations (26)-(28) and dropping the term \( J_z \) since the rotation around the beam axis is not considered, we obtain

\[ m\ddot{u} - Q_u + c_u \dot{u} = \left\{ \left( D_u \psi'\right)' \right\} \end{array} ' + \lambda (1 + u') \right\} \]  

\[ m\ddot{v} - Q_v + c_v \dot{v} = \left\{ \left( -D_v \psi'\right)' \right\} (1 + u') + \lambda v' \right\} \]  

Apply the boundary condition \( u(0, t) = 0 \) in Equation (29), we obtain

\[ u = -\int_{0}^{l} \left[ \frac{1}{2} v'^2 + \frac{1}{8} v'^4 \right] ds \]  

Apply the boundary condition \( G_u (L, t) = 0 \) in Equation (24), we obtain

\[ \lambda = \left[ \frac{1}{2} \int_{0}^{l} v'^2 \right] - m \int_{L}^{l} \left[ \frac{1}{2} v'^2 + \frac{1}{8} v'^4 ds \right] ds - \int_{0}^{l} Q_u ds \right\} \]  

Here the Lagrange multiplier \( \lambda (s, t) \) is interpreted as a force tangent to the neutral axis of the beam, i.e. the axial force, with which we can maintain the inextensionality condition.

As shown in Figure 1, the forces can be expressed by
\[ Q_v = -mg \]
\[ Q_s = \cos(\Omega t)\delta(s-L) \tag{35} \]

where \( g \left( = 9.81 \text{m/s}^2 \right) \) denotes the gravitational acceleration, \( \delta(s) \) is the Dirac function.

Substituting Equations (33) and (34) into (32) and keeping terms up to order five and using the boundary conditions \( v(0,t)=0 \), \( \dot{v}(0,t)=0 \), \( v^\prime (L,t)=0 \), we obtain the following equation of motion of the beam.

\[
m\ddot{v} + c_v \dot{v} + D_v \dddot{v} = -D_c \left( v^{(3)} + 4v'^2v^{(2)} + 2v'^2v^{(3)} + 6v'^3v^{(2)} + 8v'^3v^{(3)} + v'^4v^{(3)} \right) - \]
\[
m \left( \frac{3}{2}v'^4v^{(3)} \right) \int_0^L \left[ \left( \frac{1}{2}v'^2 + \frac{1}{8}v'^3 \right) ds \right] \frac{m}{2} \left( \frac{1}{2}v'^2 + \frac{1}{8}v'^3 \right) ds] + \]
\[
mg \left[ (s-L)v^{(2)} + v^{(3)} \right] + mg \left[ (s-L)\frac{15v'^2v^{(3)}}{2} + \frac{3v'^2}{8} + \frac{3v'^3}{8} \right] \cos(\Omega t)\delta(s-L) \tag{36} \]

The systems with multiple degrees of freedom are formulated for analysis with Galerkin Method. Then the Runge-Kutta Method is adopted to analyze the equations of motions in time domain.

**3 ILLUSTRATIVE EXAMPLE**

In this example, a steel beam with prismatic I-shaped cross section bended along the weak axis is considered. The dimensions of this beam with length \( L = 8.7 \text{ m} \) is given in Table 1.

<table>
<thead>
<tr>
<th>Type</th>
<th>Size (mm)</th>
<th>Area of Section ( A ) (cm(^2))</th>
<th>Mass per unit length ( m ) (kg/m)</th>
<th>( y-y ) Axis</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \text{I}2a )</td>
<td>320 130 9.5 15 1.5</td>
<td>67.1</td>
<td>52.7</td>
<td>459 70.6 2.62</td>
</tr>
</tbody>
</table>

Three mode functions of linear beam are used in formulating the 3-degree-of-freedom (3DOF) system with Galerkin method. The eigenvalues of the linear beam are listed in the following.

\[ \omega_1 = 63.4972 \text{ rad/s} \quad \omega_2 = 397.9591 \text{ rad/s} \quad \omega_3 = 1.1144 \times 10^3 \text{ rad/s} \]

The approximate lateral deflection of the beam is expressed by

\[ v(s,t) = \sum_{i=1}^3 q_i(t)C_i\phi_i(s) \tag{37} \]

where \( \phi_i(s) \) is the \( i \)th mode function of the beam, and \( q_i(t) \), \( C_i \) are the generalized coordinate corresponding to the \( i \)th mode and the normalizing constant, respectively.
With Galerkin method, the equations of motion in time domain are formulated and given as follows.

$$I\ddot{q} + Cq + Kq = -\frac{D}{m} NLS - NLI + \frac{F}{m} \cos(\Omega t) \quad (38)$$

in which matrix $I$ is $3 \times 3$ identity matrix; $NLS = \{NLS_1, NLS_2, NLS_3\}^T$ where $NLS_i$ is the stiffness nonlinear terms corresponding to the $i$th mode; $NLI = \{NLI_1, NLI_2, NLI_3\}^T$ where $NLI_i$ is the inertial nonlinear terms corresponding to the $i$th mode; $F = \{F_1, F_2, F_3\}^T$ where $F_i$ is the forcing term corresponding to the $i$th mode; matrix $C = diag\left(2\xi_1\omega_1^2, 2\xi_2\omega_2^2, 2\xi_3\omega_3^2\right)$ is the damping matrix where $\xi_i$ and $\omega_i$ are the damping ratio and natural frequency corresponding to the $i$th mode, respectively; matrix $K = diag\left(\omega_1^2, \omega_2^2, \omega_3^2\right)$ is the stiffness matrix; vectors $q$, $\dot{q}$ and $\ddot{q}$ are the generalized displacement, velocity, and acceleration vectors, respectively.

These three equations are coupled by the nonlinear terms and ready for numerical analysis. Runge-Kutta method is used to get the numerical solutions of Equations (38). To get the frequency-response curve (FRC), fix the amplitude of the excitation, then obtain the responses with different excitation frequency. The FRC curve of the isotropic and inextensible Euler-Bernoulli cantilever beam with nonlinearity up to fifth order are obtained and shown in Figure 4, respectively. Strong nonlinear behavior is observed in Figure 4 when the excitation frequency is around $\omega_1$.

In this paper two cases have been studied and relevant results are presented. The first case is about the investigation on the behavior of the 32a I-shaped cantilever steel beam with lateral concentrated excitation being of amplitude $25 \text{kN}$. The line with * and the line with ○ in Figure 4 represent the FRC of the beam. Only softening effect can be observed in this case since the inertial nonlinearity plays a significant role on the behavior of the vibration of the cantilever beam. When the excitation amplitude is increased to $120 \text{kN}$, as shown by the line with × and the line with ▽ in Figure 4. As both displacement and velocity are large in this case, both the hardening stiffness nonlinearity and inertial nonlinearity are strong and hence both hardening and softening effect can be observed as shown in the Figure 4. It is also observed that when the excitation frequency is close to the first natural frequency, the vibration of the cantilever beam can have both softening and hardening behavior when the excitation is strong enough. Hence we find that both softening and hardening effect can dominate the response of the motions of geometrically nonlinear cantilever in some cases. For stubby beams, the natrual frequency of the beam is pretty large which means the velocity and acceleration of the cantilever beam is large. In this case, the effects of inertial nonlinearity dominates the response behavior of the cantilever. As the excitation amplitude increases, the displacement and velocity of the cantilever also increase, which can leads to more obvious nonlinear behavior of the cantilever vibration.

4 SUMMARY AND CONCLUSIONS

The equations of motion of a cantilever beam subjected to a lateral harmonic force and axial self-weight are derived. The nonlinear dynamical system considering three mode function formulated with Galerkin method is formulated. Numerical analysis on the responses of the nonlinear dynamical system by considering fifth order nonlinear terms is conducted. It is found that the stiffness nonlinearity in the cantilever beam can lead to hardening behavior of the beam vibration. However, numerical analysis shows that the inertial nonlinearity can have softening effect. Moreover, both hardening and softening effects can be found in vibration of nonlinear cantilever beams when the excitation frequency is around the first natural frequency, depending on the amplitude of excitations and the how slender the beam is.
5 ACKNOWLEDGEMENT

This research is supported by the Research Committee of University of Macau (Grant No. MYRG138-FST11-EGK, MYRG2014-0084-FST) and the Science and Technology Development Fund of Macau (Grant No. 043/2013/A).

REFERENCES


beam with static deflection. II: Superharmonic and subharmonic resonances, Nonlinear

beam with static deflection. III: Passage through resonance, Nonlinear Dynamics, Vol. 4,
pp. 461-481.

[10] Zaretzky, C. L. and Crespo da Silva, M. R. M., 1994a, "Experimental investigation of non-
linear modal coupling in the response of cantilever beams", Journal of Sound and

Cantilever Beam with an Internal Resonance,” in Proceedings of the International Modal
Analysis Conference (IMAC), Orlando, Florida, pp. 1614-1621.

on the nonplanar autoparametric responses of an inextensible beam", in Proceedings of the

Nonlinear Flapwise and Chordwise Vibration of Cantilever Composite Beam Under Base
Excitation,” in International Journal of Structural Stability and Dynamics, Vol. 12, No. 2


York.

Quadratic manifolds for reduced-order modelling of highly flexible multibody systems

Long Wu *, Paolo Tiso #, Shobhit Jain *, Fred van Keulen *

* Faculty of Mechanical, Maritime and Materials Engineering
Delft University of Technology
Mekelweg 2, 2628CD Delft, The Netherlands
{L.Wu-1}{S.Jain-2}{A.vanKeulen}@tudelft.nl

# Institute for Mechanical Systems
ETH Zürich
LEE Bldg, M.205, Leonhardstrasse 21, 8092, Zürich
ptiso@ethz.ch

ABSTRACT
The Floating Frame of Reference (FFR) provides a natural framework for the Model Order Reduction (MOR) of flexible multibody systems. The classical reduction carried out by a Galerkin projection on a reduced basis of Vibration Modes (VMs), however, is not applicable when the elastic deformations become finite. In this contribution, we present a MOR technique based on a quadratic manifold on which the reduced solution lives. The manifold is built by an expansion of the elastic displacements for each flexible body. The quadratic terms are formed by Modal Derivatives (MDs) that properly account for the effect of the geometric nonlinearity. As opposed to classical Galerkin projection for geometrically nonlinear systems, this approach minimizes the size of the reduced order model, at the price of a more complex nonlinear system.

Keywords: Floating Frame of Reference, Model Order Reduction, Quadratic Manifold, Modal Derivatives.

1 INTRODUCTION
The floating frame of reference, which follows a mean rigid body motion of an arbitrary flexible component, is widely applied in flexible multibody dynamics, where the system components undergo large rotations [1]. Because of the inherent decoupling between rigid body motion and elastic deformation, the Floating Frame of Reference (FFR) provides a natural framework for Model Order Reduction (MOR) of flexible multibody systems. A classical linear reduction basis consisting of Vibration Modes (VMs) is very effective when the elastic deflections are small. However, when geometric nonlinearities have to be considered, VMs fail to correctly reproduce nonlinear couplings and large deflections, and therefore are of limited applicability.

In an earlier contribution [2], it was shown that a reduced basis formed with few relevant VMs can be enriched with Modal Derivatives (MDs) stemming from the sensitivity of the eigenvalue problem for free vibration with respect to modal amplitudes. This reduction correctly captures the nonlinear bending-stretching behavior associated to elastic geometric nonlinearities. While simple and effective, this approach bears the drawback of a relatively large reduced basis, as the number of obtainable MDs scales quadratically with the size of the corresponding set of VMs [3]. Consequently, the MOR loses the computational efficiency when numerous MDs have to be included in the reduced basis.

In order to overcome this drawback, we present an alternative approach based on a quadratic manifold for the reduction. For this purpose, the VMs and MDs are combined in a quadratic coordinate transformation. The proposed method exploits a second-order Taylor series expansion of the elastic deflection, with respect to a reduced set of generalized coordinates. This differs from the previous work in [2], where additional coordinates have to be built for each MD.
The proposed approach is illustrated on the basis of a two-dimensional Euler-Bernoulli beam system. The geometric nonlinearities are modelled with the von Karman kinematic relations. The reduced model with quadratic manifold is compared to both the fully nonlinear dynamics model and the reduced model with linear manifold to assess the potentiality of the proposed approach.

2 FINITE ELEMENT FORMULATION IN FLOATING FRAME

In the FFR formulation, the global motion of an arbitrary point on the element $P_j$ of body $S_i$ can be expressed as a combination of two components: the motion of the body coordinate $X_i^i Y_i^i$ and the position of points with respect to the body coordinate, as shown in Figure 1. Therefore, the global position vector $r^{ij}$ can be expressed in the FFR formulation as

$$r^{ij} = R^i + A^i \left( e^{ij}_0 + N^{ij} q_f^i \right),$$

where $R^i$ represents the location of the origin of $X_i^i Y_i^i$ with respect to global coordinates $X Y$; $A^i$ is the transformation matrix from body system $X_i^i Y_i^i$ to global system $X Y$; $e^{ij}_0$ is the undeformed position of an arbitrary point on element $P_j$ relative to $X_i^i Y_i^i$ system; $N^{ij}$ are the appropriated shape functions and $q_f^i$ is the vector of elastic nodal Finite Element (FE) displacements of body $S_i$.

![Figure 1. Coordinate definition in FFR of a generic flexible body $S_i$. The FE nodal elastic displacements are measured with respect to a body coordinate $X_i^i Y_i^i$, which is different for each flexible body.](image)

The definition of the body coordinate system $X_i^i Y_i^i$ is not unique. We adopt here the nodal fixed frame [4], where the origin of the body coordinate is fixed to one node of the body, and the $O^i X^i$ axis passes through the end node of the beam structure.

Since the focus of this work is the reduction of the elastic contribution in the FFR, we restrict ourselves here to a single body system in order to better illustrate the proposed method. The extension to multibody system is straightforward. Therefore, the superscript $i$ can now be omitted for clarity.
The dynamic equations for flexible system in 2D plane can be obtained with the Lagrange equations for constrained bodies

\[
\frac{d}{dt} \left( \frac{\partial T}{\partial \dot{q}} \right)^T - \left( \frac{\partial T}{\partial q} \right)^T + \left( \frac{\partial U}{\partial q} \right)^T + C^T q_e = 0,
\]

where \( T \) and \( U \) are the kinetic energy and strain energy; \( C_q \in \mathbb{R}^{d,3+n} \) is the constraint Jacobian matrix; \( q_e \) is the vector of Lagrange multipliers; \( q_e \in \mathbb{R}^{3+n} \) is the vector of externally applied forces; \( t \) is the time; \( d \) is the number of constraint equations; \( q \) is the vector of system generalized coordinates, which could be expressed as

\[
q = \begin{bmatrix} q_r \\ q_f \end{bmatrix} = \begin{bmatrix} R \\ \theta \\ q_f \end{bmatrix},
\]

where \( q_r \in \mathbb{R}^3 \) represents the displacement and orientation of body coordinates, \( q_f \in \mathbb{R}^n \) refers to the elastic displacement in body coordinates, \( \theta \) is the rotational coordinates that describe the angular rotation of \( X^rY^r \) with respect to \( XY \) and \( n \) is the number of elastic FE degrees of freedom (dofs).

The inertia coupling between the body coordinates \( q_r \) and elastic coordinates \( q_f \) generates inertial terms which are configuration and velocity dependent

\[
\frac{d}{dt} \left( \frac{\partial T}{\partial \dot{q}} \right)^T - \left( \frac{\partial T}{\partial q} \right)^T = M(q) \ddot{q} - Q_e(q, \dot{q}),
\]

where \( M \in \mathbb{R}^{3+n,3+n} \) is the system mass matrix, and \( Q_e \in \mathbb{R}^{3+n} \) is the quadratic velocity vector.

The elastic internal force vector is here directly derived from the differentiation of strain energy \( U \) with respect to the generalized coordinates

\[
Q = \frac{\partial U}{\partial q} = \begin{bmatrix} \frac{\partial U}{\partial q_r} \\ \frac{\partial U}{\partial q_f} \end{bmatrix} = \begin{bmatrix} 0 \\ Q_{nl} \end{bmatrix},
\]

where \( Q_{nl} \in \mathbb{R}^n \) is a third-order polynomial of the elastic nodal displacement \( q_f \). Finally, the equation of motion (2) could be compactly written as

\[
M(q) \ddot{q} + Q(q) = Q_e + Q_e(q, q) - C^T_q(q) \lambda.
\]

Equation (6) can be conveniently expressed in a partitioned form:

\[
\begin{bmatrix} M_{rr} & M_{rf} \\ M_{fr} & M_{ff} \end{bmatrix} \begin{bmatrix} \ddot{q}_r \\ \ddot{q}_f \end{bmatrix} + \begin{bmatrix} 0 \\ Q_{nl} \end{bmatrix} = \begin{bmatrix} (Q_e)_r \\ (Q_e)_f \end{bmatrix} + \begin{bmatrix} (Q_e)_r \\ (Q_e)_f \end{bmatrix} - \begin{bmatrix} C^T_{qr} \\ C^T_{qf} \end{bmatrix} \lambda,
\]

where subscripts \((*)_r\) and \((*)_f\) refer to the body and elastic coordinates, respectively.

### 3 Model Order Reduction Through Projection

The size of the system of equations (7) could be largely reduced via a Galerkin projection onto a small subspace spanned by a reduced basis. In structural dynamics, VMs from an eigenvalue analysis are a typical choice for the construction of such basis, but other possibilities also exist. When applied to nonlinear analysis, this approach gives poor accuracy since the modal basis is configuration dependent. In general, a basis upgrade is required during the time integration to suitably consider the effect of the nonlinearities during the time integration.

In previous contributions [2], we showed that enriching the reduced basis with MDs greatly improves the accuracy of the reduced model. However, the number of MDs one can obtain grows quadratically with the size of the underlying set of VMs. In this work, we propose a nonlinear manifold for the reduction that keeps the size of the reduced order model equal to the number of retained vibration modes.
3.1 Nonlinear displacement expression

When the elastic displacements $q_f$ cannot be considered as small, the VMs change with respect to the configuration. Therefore, the displacement vector $q_f$ can be expressed as

$$ q_f = X_f(\eta) \eta, \quad (8) $$

where the dependence of the modal basis $X \in \mathbb{R}^{n \times m}$ on the modal amplitude $\eta \in \mathbb{R}^m$ is highlighted. Einstein summation convention is adopted from here on, i.e., repeated indices in the subscript are implicitly summed over.

We now expand $q_f$ in Taylor series around the equilibrium position $y|_{eq}$ to obtain

$$ q_f = \left. \frac{\partial q_f}{\partial \eta_i} \right|_{eq} \eta_i + \frac{1}{2} \left. \frac{\partial^2 q_f}{\partial \eta_i \partial \eta_j} \right|_{eq} \eta_i \eta_j + \ldots \quad (9) $$

The derivatives of the elastic displacement vector $q_f$ with respect to the modal amplitudes $\eta_i$ can be computed from equation (8) as

$$ \frac{\partial q_f}{\partial \eta_i} = X_i + \frac{\partial X_i}{\partial \eta_i} \eta_j \quad (10) $$

and

$$ \frac{\partial^2 q_f}{\partial \eta_i \partial \eta_j} = \left. \frac{\partial^2 \tilde q_f}{\partial \eta_i \partial \eta_j} \right|_{eq} \eta_i + \frac{\partial^2 \tilde X_i}{\partial \eta_i \partial \eta_j} \eta_j. \quad (11) $$

All the derivatives in equation (10) and (11) can be calculated at the equilibrium position as

$$ \Phi_i = \left. \frac{\partial q_f}{\partial \eta_i} \right|_{eq} = X_i|_{eq} \quad (12) $$

and

$$ \Theta_{ij} = \left. \frac{\partial^2 q_f}{\partial \eta_i \partial \eta_j} \right|_{eq} = \left. \frac{\partial X_i}{\partial \eta_j} \right|_{eq} + \left. \frac{\partial X_j}{\partial \eta_i} \right|_{eq}, \quad (13) $$

where $\Phi \in \mathbb{R}^{n \times m}$ are the VMs around the equilibrium position and $\Theta \in \mathbb{R}^{n \times m \times m}$ are the MDs.

With the notation $\Phi_i$ and $\Theta_{ij}$, we indicate the column vectors corresponding to the first-index of $\Phi$ and $\Theta$, i.e., $\Phi_i = \Phi(:, i)$ and $\Theta_{ij} = \Theta(:, i, j)$.

By neglecting all the higher order terms in the expansion (9), the elastic displacement $q_f \in \mathbb{R}^n$ here lives on a quadratic manifold $\Gamma(\eta) : \mathbb{R}^m \mapsto \mathbb{R}^n$, as

$$ q_f = \Gamma(\eta) = \Phi_i \eta_i + \frac{1}{2} \Theta_{ij} \eta_i \eta_j. \quad (14) $$

The modal derivatives can be computed from the eigenvalue problems for geometric nonlinear cases, obtained from [5]:

$$ (K_{nl}(\eta) - \omega_i^2(\eta)M_{ff}) X_i = 0, \quad (15) $$

where $K_{nl}$ is the tangent stiffness matrix. The derivatives of equation (15) with respect to $\eta_j$ give

$$ \left( K_{nl} - \omega_i^2 M_{ff} \right) \frac{\partial X_i}{\partial \eta_j} = - \frac{\partial K_{nl}}{\partial \eta_j} X_i + \frac{\partial \omega_i^2}{\partial \eta_j} M_{ff} X_i. \quad (16) $$

Evaluating equation (16) around equilibrium position gives:

$$ \left( K_l - \omega_i^2 M_{ff} \right) \left. \frac{\partial X_i}{\partial \eta_j} \right|_{eq} = - \left. \frac{\partial K_{nl}}{\partial \eta_j} \right|_{eq} \Phi_i + \left. \frac{\partial \omega_i^2}{\partial \eta_j} \right|_{eq} M_{ff} \Phi_i, \quad (17) $$

425
where $K_L$ is the linear stiffness matrix. The computation of equation (17) is not straightforward, because the term $(K_L - \omega^2 M_{ff})$ is singular around the equilibrium position. Therefore, it is generally augmented with the derivatives of the normalization condition to constrained the amplitude and handle to singularity

$$
\Phi_i^T M_{ff} \frac{\partial X_i}{\partial \eta_j} \bigg|_{eq} = 0, \quad \forall i, j = 1, \cdots, m.
$$

Therefore, the modal derivatives can be calculated as

$$
\begin{bmatrix}
K_L - \omega^2 M_{ff} & -M_{ff} \Phi_i \\
-(M_{ff} \Phi_i)^T & 0
\end{bmatrix}
\begin{bmatrix}
\frac{\partial X_i}{\partial \eta_j} \bigg|_{eq} \\
\frac{\partial \omega^2}{\partial \eta_j} \bigg|_{eq}
\end{bmatrix} =
\begin{bmatrix}
- \frac{\partial K_L}{\partial \eta_j} \bigg|_{eq} \\
0
\end{bmatrix} \Phi_i.
$$

### 3.2 Reduction with linear projection

In previous research [6, 7], it has been shown that an efficient reduced basis for the nonlinear problems can be formed by combining dominant VMs with some corresponding MDs, so that the dynamic contribution can be approximated by both the VMs and MDs as

$$q = \begin{bmatrix} q_r \\ q_f \end{bmatrix} = \begin{bmatrix} q_r \\ \Phi_i \eta_i + \Theta_{ij} \xi_k \end{bmatrix} = \begin{bmatrix} I_{3 \times 3} & 0 \\ 0 & \Phi \end{bmatrix} \begin{bmatrix} q_r \\ \eta \\ \xi \end{bmatrix},
$$

where $\Theta_{ij}$ is formed with columns of $\Theta_{ij}$, and $\xi$ are independent coordinates associated to the modal derivatives. Equation (20) can be written more compactly

$$q = \Psi \chi,
$$

where $\Psi \in \mathbb{R}^{3+n, 3+m+r}$ is the tangent space associated to the linear manifold, $\chi \in \mathbb{R}^{3+m+r}$ is the generalized modal coordinates, and $r$ is the number of MDs that could be calculated with $m$ given VMs. By inserting the linear manifold approximation (21) into the governing equation (6), and projecting on the tangent subspace $\Psi$, the reduced equation of motion can be obtained as

$$\bar{M}(\chi) \ddot{\chi} + \bar{Q}_m(\chi) = \bar{Q}_c + \bar{Q}_r (\chi, \dot{\chi}) - \bar{C}_q(\chi) \lambda,
$$

where $\bar{M} \in \mathbb{R}^{3+m+r, 3+m+r}$, $\bar{Q}_m \in \mathbb{R}^{3+m+r}$, $\bar{Q}_c \in \mathbb{R}^{3+m+r}$, $\bar{Q}_r \in \mathbb{R}^{3+m+r}$ and $\bar{C}_q \in \mathbb{R}^{d, 3+m+r}$. The details can be found in [2]. While simple and effective, this approach bears the drawback of the largely increasing size of reduced basis. It can be shown that the modal derivatives are symmetric, i.e. $\Theta_{ij} = \Theta_{ji}$. Thus for a given set of $m$ VMs, $r = m(m+1)/2$ MDs can be calculated. Therefore, the number of MDs grows quadratically with respect to $m$. The next section will present an alternative approach to eliminate this drawback.

### 3.3 Reduction with nonlinear projection

Based on the expansion (14), the generalized displacement $q$, velocity $\dot{q}$ and acceleration $\ddot{q}$ can be written as

$$q = \begin{bmatrix} q_r \\ \Phi_i \eta_i + \frac{1}{2} \Theta_{ij} \eta_i \eta_j \end{bmatrix},
$$

$$\dot{q} = \begin{bmatrix} \dot{q}_r \\ \dot{\Phi}_i \eta_i + \frac{1}{2} \dot{\Theta}_{ij} \eta_i \eta_j + \frac{1}{2} \Theta_{ij} \eta_i \dot{\eta}_j \end{bmatrix},
$$

and

$$\ddot{q} = \begin{bmatrix} \ddot{q}_r \\ \ddot{\Phi}_i \eta_i + \frac{1}{2} \ddot{\Theta}_{ij} \eta_i \eta_j + \ddot{\Theta}_{ij} \eta_i \dot{\eta}_j + \frac{1}{2} \Theta_{ij} \eta_i \ddot{\eta}_j \end{bmatrix}.
$$
The tangent space $V(\eta) \in \mathbb{R}^{3+n+3+m}$ for the generalized dofs in FFR can then be computed as

$$V(\eta) = \begin{bmatrix} I_{3 \times 3} & 0 \\ 0 & \frac{\partial \Gamma}{\partial \eta} \end{bmatrix} = \begin{bmatrix} I_{3 \times 3} & 0 \\ 0 & \Phi + \Theta \eta \end{bmatrix}. \tag{26}$$

By inserting the quadratic manifold approximation (23), (24), and (25) in the governing equation (6), and projecting on the tangent subspace (26), the reduced equation of motion can be obtained as

$$V^T M(q_r, \eta) \ddot{q} + V^T Q_{nl}(\eta) = V^T Q_e + V^T Q_v(q_r, \dot{q}_r, \eta, \dot{\eta}) - (C(q_r, \eta)V)^T \lambda, \tag{27}$$

or more compactly

$$\tilde{W}(q_r, \dot{q}_r, \eta, \dot{\eta}) + \tilde{Q}_{nl}(\eta) = \tilde{Q}_e(\eta) + \tilde{Q}_v(q_r, \dot{q}_r, \eta, \dot{\eta}) - \tilde{C}_q^T(q_r, \eta)\lambda, \tag{28}$$

where $\tilde{W} \in \mathbb{R}^{3+m}, \tilde{Q}_{nl} \in \mathbb{R}^{3+m}, \tilde{Q}_e \in \mathbb{R}^{3+m}, \tilde{Q}_v \in \mathbb{R}^{3+m}$ and $\tilde{C}_q \in \mathbb{R}^{d,3+m}$. $\tilde{W}$ is the reduced inertia force, which is very complex owing to the nonlinear projection in floating frame. Note that the system of reduced equations is now of size $3 + m$, as compared to the Galerkin-reduced projection of size $3 + m + r$.

To clearly illustrate how the reduction approach works in the FFR, a rotating beam example is shown in Figure 2, where the first 2 VMs and corresponding 3 MDs are included. From Figure 2 we can see that the first few VMs feature bending displacements only, while the corresponding MDs, which describe the bending-stretching coupling nonlinearities, exhibit axial displacement only.

![Figure 2. The MOR for the rotating beam in FFR](image-url)
4 Numerical example

The reduced order model with linear manifold (LMROM) and reduced order model with quadratic manifold (QMROM) in FFR are tested in this section, and compared to full model (FM), on a 2D beam example. We introduce here a slender rubber beam of length \( L = 1 \) m, circular cross section of 5 mm radius. The beam is fixed to its left end via a revolute joint and exposed to the gravitational field. The rubber material parameters are chosen as \( E = 5 \times 10^6 \) N.m\(^{-2}\), \( \rho = 1.1 \times 10^3 \) kg.m\(^{-3}\). This example has already been discussed by other authors [8, 9].

The implicit Newmark scheme is applied, with parameters \( \alpha = 0.5 \) and \( \beta = 0.25 \). The beam is initially at rest in horizontal position. The analysis is performed for a time interval of 1 seconds with a fixed time step \( \Delta T = 0.001s \).

The time evolution of the elastic beam centerline during the first 1 second is shown in Figure 3. Figure 3(a) shows a good agreement between FM solution and the one using LMROM, composed of first 4 VMs and corresponding 10 MDs (indicated as 4VMs+10MDs). Similarly, Figure 3(b) exhibits a good agreement between the QMROM, based on first 8 VMs, and FM.

It can be seen from Figure 3 that the MDs are able to capture the second-order nonlinear effects, and as a result accurately describe the nonlinear dynamic behavior, using both LMROM and QMROM.

To offer a clear view, the root mean square (RMS) error \( \varepsilon_{RMS} \) relative to the entire displacement vector is shown in Figure 4, calculated as

\[
\varepsilon_{RMS} = \frac{1}{n} \left[ \sum_{i=1}^{n} (u_i - \tilde{u}_i)^2 + \sum_{i=1}^{n} (w_i - \tilde{w}_i)^2 \right],
\]

where \( u_i, w_i \) and \( \tilde{u}_i, \tilde{w}_i \) are the horizontal and vertical components of the node displacement from the full and reduced models.

Results are shown in Fig.4. The QMROM converges to the reference solution when the modal coordinates increase from 4VMs to 8VMs and 14VMs. The QMROM outperforms the LMROM of the same size. Compared to the LMROM where 14 modal coordinates (4VMs+10MDs) are included in total, the QMROM can generate much better results when only 8 modal coordinates need to be considered.

The additional computational cost associated to the increased complexity of the nonlinearity is currently under investigation and will be addressed by future work.

5 CONCLUSIONS

In this paper, a quadratic mapping combining the linear VMs and corresponding MDs was proposed for the reduction of flexible multibody systems with FFR form. The quadratic reduced manifold results in a configuration dependent tangent space on which the full system of equations is projected. As a result, the reduced order model bears extra convective terms.

In the proposed approach, the number of modal coordinates is equal to the number of the basis vectors spanning the initial subspace. Therefore, compared to the linear manifold, where each MD is associated with an independent dof in the reduced basis, the proposed method here allows to further reduce the size of dynamic system, at the price of a more complex nonlinear system. This issue is currently under investigation.

A numerical example, featuring large rigid rotation and moderate elastic deflection, is shown to illustrate the proposed approach. For reduced models of the same dimensions, the quadratic manifold reduction showed to improve the results obtained with the linear manifold in FFR form. Further investigations should be conducted to more realistic numerical examples of multibody systems.
Figure 3. Time evolution of a swinging rubber beam
Figure 4. RMS error of the different reduction techniques with different ingredients for basis

REFERENCES


Relative and absolute finite element nodal coordinates in the dynamics of large flexible structures subject to seismic excitations

Evtim V. Zahariev*, Venelin S. Jivkov#, Ivan D. Danchev#

* Institute of Mechanics
Bulgarian Academy of Sciences
Acad. G. Bonchev St.bl. 4
1113 Sofia, Bulgaria
evtimvz@bas.bg

# Mech. and Mach. Theory Dept.
Technical University - Sofia
St. Kliment Ohridsky St. 8
1000 Sofia, Bulgaria
jivkov{dantchev}@tu-sofia.bg

ABSTRACT
In the paper general approach to dynamics simulation of rigid and flexible multibody systems based on relative and absolute coordinates of finite elements is proposed. The element nodes are considered moving coordinate systems that enables their motion coordinates to be regarded as these of rigid bodies. Relative coordinates are successfully used for open branch multibody system simulation as single beam structures or systems compiled of many consequently connected adjacent elements as, robots and manipulators, large cantilever beam and etc. Finite elements in absolute coordinates are to be applied for complex structures with many adjacent flexible elements that form closed chains and possess many dependent coordinates.

In the paper a novel method of finite elements in absolute coordinates is discussed. The flexible elements and their node coordinate systems are considered free objects in space which motion is restricted by the elastic forces of the adjacent flexible elements. So, no kinematic restrictions, respectively algebraic equations, are imposed to the dynamic equations and the dynamic model is presented as Ordinary Differential Equations. Incremental approach for definition of system configuration during its global motion is applied, which avoids singularity of the large rotations.

The method is applied for dynamics simulation of large flexible structures subject to seismic excitation. Force and acceleration approaches for foundation disturbances are discussed. The simulation of the seismic excitation is performed taking into account the specific for the region ground accelerations, which are reonomic constraints imposed on the dynamic equations.

Keywords: multibody dynamics, finite elements, relative coordinates, absolute coordinates.

1 INTRODUCTION
The methods of the finite element theory (FET) [1] have been constantly developing to successfully solve the up-to-date tasks in structural statics. The problem of deriving the dynamics equations of flexible bodies using finite element discretization is really topical. Many books and papers [2, 3] discuss the problems that could arise using the finite element methodology and commercially available software (NASTRAN, ANSYS, ABAQUS) for simulation of nonlinear phenomena. It is pointed out that the FET approach cannot effectively simulate nonlinear effects of high velocity global motion in space and large flexible deformations.

The finite element theory proposed the theoretical basis for profound numerical analysis of complex flexible structures. The methodology for discretization of the continuum and the theory for polynomial approximation of the form of the deformations, and on this basis computation and distribution of the mass and stiffness properties of the flexible elements to discrete objects (nodes), actually, have been used afterward for development of various methods for static and dynamic analysis, even in the multibody system dynamics (MDS) simulation of rigid and flexible bodies. Attempts had been made for direct application of FET for dynamics simulation of systems that undergo global motion in space superimposed by flexible deflections. In the recent decades the scientists successfully developed up-to-date methods for dynamics analysis of rigid and flexible mechanical systems setting the basis of multibody system methodology.
Actually, the scientists use the theoretical rules of the FET for deriving the basic properties of the discretized elastic bodies but, on the other hand, they strive to escape from the motion parameters of the FET in order to catch the nonlinear effects of the large displacements and deformations. It was proven [4 – 6] that using the FET approach the dynamic equations are free of centrifugal and Coriolis accelerations and the corresponding inertia forces.

Khan and Anderson [7] proposed the most recent classification of the methods for MBS dynamics. The survey is related to the so called Divide-and-Conquer algorithm used and updated for their applications. Further in the paper they used and applied the Floating Frame of Reference (FFR) and Absolute Nodal Coordinate Formulation (ANCF) approaches for realization of numerical procedures. Both methods are the most often used and effective for multibody system dynamics simulation.

The FFR approach was developed [4, 8, 9] and has been successfully applied for dynamics simulation of large spatial motion and rotation of flexible multibody systems. This approach is also used for application of the modal coordinates. The opinion of the authors is that FFR and its up-to-date developments is the most often used method in multibody system dynamics.

In recent decades the novel method of ANCF has been rapidly developing [10, 11]. This approach strives to solve many complicated problems of the dynamics of flexible multibody systems. It is based on the theory of the curvilinear coordinates. The slopes of the nodes in the tangential line or plane of the space curve or shell are applied as coordinates. The method proposes important advantages mainly in large rotation simulation and mass matrices formulation. Review on the ANCF method for large deformation dynamics simulation of flexible multibody systems was presented by Gerstmayr et al. [12]. Both methods, ANCF and FFR, are the major methods developed and used at the Department of Mechanical Engineering, University of Illinois at Chicago.

As it was said above the FET approach does not take into account the geometric stiffening. The dynamic equations derived on that basis do not include the Coriolis and the centrifugal accelerations. Shabana [13] and Meijard [14] derived generalized Newton – Euler dynamic equations for the case of FFR formulation. The equations represent the inertia forces as function of the velocities and accelerations of the global and flexible coordinates. Zahariev [15] derived generalized Newton-Euler dynamics equations for the rigid and flexible bodies for which the kinetic energy is quadratic form of the velocities. The equations are applicable for the flexible elements of the FET. The inertia forces are with respect to the quasi velocities and accelerations and are invariant to the kind of the coordinates.

In [16] a method of Finite Elements in Relative Coordinates (FERC) based on the FET was proposed for dynamics simulation of large flexible structures. Using the generalized Newton-Euler dynamic equations nonlinear effects and the geometrical stiffening are successfully simulated. This method is fully compatible with the finite element discretization in FET. But using this approach for complex structures with many mutually connected adjacent flexible bodies one could experience difficulties that mainly consists in its program system realization.

In the paper a method of Finite Elements in Absolute Coordinates (FEAC) is proposed for dynamics simulation of complex structures with many adjacent flexible elements that form closed chains and possess many dependent coordinates. The method is natural continuation of the investigations related to the FERC [16] and strives to cope with the problems that arise in dynamics simulation of large flexible structures with many degrees of freedom (dof) and dependent coordinates like structures and mechanical devices built as complex girders and etc.

The method FEAC is applied for simulation of structures subject to seismic excitations. Acceleration and force methods are discussed for dynamics simulation of earthquake structure response. Acceleration approach consists in registration of the motion of a structure foundation by accelerographs and normally consists in three orthogonal components of the ground acceleration. The velocity and displacement of the ground are obtained integrating the data of the accelerograms. In [17] the accelerations of the foundation are reonomic constraints for the dynamic equations.
Investigation of the forces loading the basement is a natural way for simulation of the structure response because of earthquakes [18]. This task is related to the soil-structure interaction investigation. Computation of the forces loading the foundation as a function of the relative motion ground – basement is the first stage for the numerical simulation process.

Both methods for seismic excitation simulation could be applied using the FEAC approach proposed in the paper. Test examples for verification of the results obtained using the methods of FERC and FEAC are presented. An example of dynamics simulation of large wind power generator of rigid and flexible bodies taking into account high speed rotor rotation with eccentricity and seismic excitation is also presented in the paper.

2 FINITE ELEMENTS, RELATIVE AND ABSOLUTE NODAL COORDINATES

To understand the methods of the finite elements in relative and absolute nodal coordinate formulation using multibody system methodology some explanations about the kinematic coordinates will be presented here. Since the basic relations and deductions as discretization of the flexible elements, the forms of the deformations, as well as, the methods for computation of the mass and stiffness matrices are well known for the readers [1, 19] no special attention will be paid to that procedures. In Figure 1 a non-isoparametric space beam element (with index \( i \)) is presented. This finite element is relatively simple for explanation but, at the same time, presents the most common six dof space deflections of the nodes. In Figure 1(a) the beam and small translations \( \Delta x_{i,k} = [\Delta x_{i,k} \Delta y_{i,k} \Delta z_{i,k}] \) and rotations \( \theta_{i,k} = [\theta_{i,k} \theta_{i,k} \theta_{i,k}] \), \( i = 1,2 \) of its nodes (indices \( i,1 \) and \( i,2 \) – the first and second nodes of element \( i \)) with respect to the initial not deformable element coordinate system \( X_0 Y_0 Z_0 \) are depicted [1, 19]. With the superscript “\(^{t}\)” a matrix transpose is denoted. The underlined notations denote initial configurations. With bold characters matrices are pointed out. For points and objects italic notations are used. It should be said that in FET these deflections are with respect to the element reference frame in its initial position and no rotation of the element coordinate system is taken into account. Of course, coordinate system transformation with respect to the absolute reference frame \( X_0 Y_0 Z_0 \) of mass and stiffness matrices is considered but for the structure initial configuration only. Since the rotational deflections are with respect to three orthogonal axes the cross-sections in the end points (nodes) are not points but coordinate systems. In [20] the regulations for coordinate system orientation of moving elements and nodes for different iso- and non-isoparametric elements are presented. For example, for a beam its element coordinate system coincides with that of node \( i,1 \).

\[ \Delta x_{i,k} = [\Delta x_{i,k} \Delta y_{i,k} \Delta z_{i,k}] \]

\[ \theta_{i,k} = [\theta_{i,k} \theta_{i,k} \theta_{i,k}] \]

\( i = 1,2 \)

\( \Delta x_{i,k} \)

\( \theta_{i,k} \)

\( \Delta x_{i,k} \)

\( \theta_{i,k} \)

\( i,1 \)

\( i,2 \)

\( X_0 \)

\( Y_0 \)

\( Z_0 \)

\( X_0 \)

\( Y_0 \)

\( Z_0 \)

\( i,1 \)

\( i,2 \)

\( X_0 \)

\( Y_0 \)

\( Z_0 \)
As a result of the node deflections the element coordinate system $X_iY_iZ_i$, respectively the node $i,1$ coordinate system $X_{i,1}Y_{i,1}Z_{i,1}$, moves to a new position defined by the radius – vector $\mathbf{p}_i$, that could be easily estimated adding together the initial position of element coordinate system origin, radius – vector $\mathbf{p}_i = \mathbf{p}_{i,1}$, and the node $i,1$ deflections $\Delta_{i,1}$. The same holds for the radius – vector $\mathbf{p}_{i,2}$, taking into account the initial position of node $i,2$ coordinate system origin, radius – vector $\mathbf{p}_{i,2}$, and the node $i,2$ deflections $\Delta_{i,2}$. The node deflections of the $i$-th element are set in a $12 \times 1$ matrix – column (vector) $\Theta_i = [\Theta_{i,1} \Theta_{i,2}] = [\Delta_{i,1} \Delta_{i,2} \Theta_{i,1} \Theta_{i,2}]$.

In Figure 1(b) the element $i$ is not drawn but only the coordinate systems $i,1$ and $i,2$ of its nodes are shown in their initial and final configurations. Since six possible deflections are assumed (translations and rotations), they are actually free objects, respectively coordinate systems that implement free space motion restricted by the elastic forces only. The elastic forces $F_i$ and torques $L_i$ in the first node, as well as, $F_{i,2}$, $L_{i,2}$ in the second node, which are $3 \times 1$ matrix – vectors are set in a $12 \times 1$ matrix – column $\Xi_i = [F_{i,1} L_{i,1} F_{i,2} L_{i,2}]$ arranged in the same order as the corresponding elements of $\Theta_i$.

### 2.1 Finite elements in relative coordinates

Actually, the position of the element coordinate system $X_iY_iZ_i$ in the deformed configuration coincides with the coordinate system $X_{i,1}Y_{i,1}Z_{i,1}$ of node $i,1$. So, the deflections of the flexible element could be considered as free motion of the element coordinate system and the flexible deflections of the second node with respect to it. These are the flexible deflections in relative coordinates, i.e., relative to the element coordinate system. On this base the FERC approach is defined and one could realise that it is similar to FFR approach but using small flexible deflections defined in FET as parameters.

In Figure 2 the beam element in relative coordinates is presented. In Figure 2 (a) the beam element in the configuration as this of Figure 1 (a) is depicted including the length $L_i$ of the beam (along axis $X$) and the small deflections $\Delta_{i,2}/i$ of node $i,2$ relative to the beam coordinate system $i$. In the right subscript the symbol “$)$” serves as an arrow pointing out that the coordinate system with index $i,2$ is with respect to coordinate system $i$. The right symbol of the subscript could be omitted if it points the absolute reference frame (index “0” – zero). Obviously, the beam configuration is quite the same and the fact that the flexible deflections are read relative to the moving element coordinate system does not change the regulations for FET discretization and for computation of the mass and stiffness matrices. The position of node $i,2$ coordinate system relative to the coordinate system of the moving beam is presented by $4 \times 4$ homogeneous transformation matrix $T_{i,2}/i$, i.e.:

![Figure 2. Finite beam element in relative coordinates](image-url)
\[ T_{i,2} = \begin{bmatrix} 3.3 \tau_{i,2} & 3.1 \rho_{i,2} \\ \end{bmatrix} = \begin{bmatrix} 1 - \theta_{i,2} & \theta_{i,2} & L_i + \Delta y_{i,2} \\ \theta_{i,2} & 1 - \theta_{i,2} & \Delta y_{i,2} \\ 0 & 0 & 0 \\ \end{bmatrix} \begin{bmatrix} 1 & -q_{i,6} & q_{i,5} & L + q_{i,1} \\ -q_{i,6} & 1 & -q_{i,4} & q_{i,2} \\ -q_{i,4} & 1 & q_{i,3} \\ 0 & 0 & 0 & 1 \\ \end{bmatrix} \]

which for small rotational deflections does not include trigonometric functions. With this assumption the homogeneous transformation matrix is linear for the relative coordinates \( q_{i,j}, j = 1,2, ... \) of element \( i \) that include the small translational and rotational flexible deflections. \( 3.3 \tau_{i,2} \) is a 3x3 rotational matrix; the left superscripts point out (used if needed) the matrix dimensions (rows, columns).

Since the beam coordinate system is moving the flexible deflections of node \( i,1 \) relative to the element coordinate system are zero and the matrix vector of the beam relative coordinates \( q_i \) is

\[ \Theta_{i,j} = \begin{bmatrix} \Delta y_{i,2} \\ \theta_{i,2} \\ \end{bmatrix} = q_i = \begin{bmatrix} q_{i,1} & q_{i,5} & \ldots & q_{i,6} \end{bmatrix} \]

(see Equation (1)). As well as, the beam mass and stiffness matrices are constant for these parameters.

### 2.2 Finite elements in absolute coordinates

In Figure 1(b) the coordinate systems of the beam element and its nodes are depicted without beam shape and configuration. As we said above the beam coordinate system and that of node \( i,1 \) coincide and their relative position does not change during the beam motion. Loading the nodes with the inertia, elastic and external forces, just like free rigid bodies they will implement, for a small increment of time, small increments of the motion parameters. Six are the minimal number of the coordinates that define the position as of a rigid body, so of a node. When a flexible body implements global motion in space overlapped with flexible deflections even small increments of the node coordinates with respect to their initial position do not present the flexible deformation of the element. The problem how to use the FET for that case and large flexible deflection consists in solution of the following two tasks:

- definition of proper node reference frame coordinates for the case of small increments of time, respectively, for small increments of that coordinates;
- estimation of the magnitudes of the elastic deformations.

The solution of the first problem consists in selection of node reference frame coordinates compatible with the coordinates used in FET. The solution of the second problem consists in separation of the flexible deflections from the global motion of the node coordinate systems.

In Figure 3, similar to Figure 1 (b), the coordinate systems of nodes \( i,1 \) and \( i,2 \), \( X_{i,k} Y_{i,k} Z_{i,k} \) : \( i = 1,2 \), are depicted in their final configuration. The node coordinate system initial positions are presented by the same but underlined notations. FET claims for the small deflections to be read along the absolute coordinate system axes and with respect to the system initial configuration.

In Figure 3 (a) these deflections are presented by arrows and in Figure 3 (b) they are presented by virtual translational and rotational joints. This presentation is one and the same but gives proper comparison to the free moving objects, constraint kinematics, joint coordinates, as well as, presents the generality of the methodology proposed in the paper.

The homogeneous transformation matrices \( T_{i,1}, T_{i,2} \) of the absolute position of the node coordinate systems \( i,1 \) and \( i,2 \), respectively, is expressed by the matrices \( T_{i,1,0}, T_{i,2,0} \) of their initial configuration and the transformation matrix because of the small node deflections, i.e.:
include also the global motion of the nodes. To define the flexible deformations
to be small, the rotations could be presented without

similarly to the FERC approach and taking into account that the relative deformations of node 1

are zero, the flexible deformations of the element \( i \) relative to its own reference frame are

appear explicitly in matrix \( T_{i,2}i \) without trigonometric functions.

To realize the numerical algorithm with the proposed coordinates the integration process should
be realized with small increments of time, respectively, with small increments of the nodal
coordinates. For this purpose on every step of the time increment the new configuration of the
system is the initial configuration for the next iteration and the absolute nodal coordinates start
again with zero values. That process is not disadvantage since during the numerical integration of the dynamic equations the new positions of the coordinate systems are to be recalculated and the new mass and stiffness properties of the bodies are to be computed. Detailed statements of the incremental procedure are presented in [20].

3 ELASTIC AND INERTIA FORCES

In the paper special attention is paid to the elastic and inertia forces loading the nodes of flexible elements. As it could be seen above the relative flexible deformations of nodes with respect to the element coordinate system, \( \Theta_{i,j} \), play different roles for FERC and FEAC. For the FERC they are coordinates of motion or generalized coordinates, while in FEAC the motion coordinates are the node deflections in absolute reference frame. But for both methods the relative flexible deformations for the first node, in which the moving coordinate system of the element is located, are equal zero. As it was shown as for FERC and FEAC the matrix-vector \( \Theta_{i,j} \) is compiled without the deformations in the first node and is used for computation of the elastic forces loading both element nodes, i.e.:

\[
\Xi_{i,j} = -K_{i,j} \cdot \Theta_{i,j},
\]

where for a beam element \( K_{i,j} \) is a 12×6 matrix compiled from the last six columns of the conventional FET stiff matrix of element \( i \) relative to its own beam coordinate system. The first six elements of \( \Xi_{i,j} \) are the forces and torques loading the coordinate system of node \( i,1 \) and the last six elements are the forces and torques loading node \( i,2 \).

Taking into account that the nodes are coordinate systems and summarizing the results for the FERC and FEAC methodologies one concludes that the dynamics analysis of multibody systems of rigid and flexible bodies could be analyzed by a common methodology. As it was discussed above one of the major problems for application of the FET in dynamics of rigid and flexible multibody systems it is definition of the node accelerations, respectively, the inertia forces. In [20] generalized Newton-Euler dynamic equations for rigid and flexible bodies discretized using the FET are derived. The inertia forces in the nodes of the flexible elements are expressed with respect to the quasi velocities and accelerations and are independent of the kind of coordinates. The notations for the quasi-velocities and accelerations of node \( i,k \) are \( \dot{\Theta}_{i,k} = \left[ \dot{v}_{i,k} \quad \dot{\omega}_{i,k} \right] \), \( \ddot{\Theta}_{i,k} = \left[ a_{i,k} \quad \ddot{e}_{i,k} \right] \) where \( \dot{v}_{i,k} \) and \( a_{i,k} \), respectively, \( \dot{\omega}_{i,k} \) and \( \ddot{e}_{i,k} \), are the node linear and angular velocities. For a flexible element \( i \) with two nodes (indices 1 and 2), and 12×12 mass matrix \( M \) the inertia forces and torques, 12×1×1 matrix vector \( F_i = \left[ p_{i,1} \quad \Phi_{i,1} \quad p_{i,2} \quad \Phi_{i,2} \right] \), is as follows:

\[
F_i = M_i \cdot \begin{bmatrix} \Theta_{i,1} \\ \Theta_{i,2} \end{bmatrix} + \begin{bmatrix} \omega_{i,1}^x & 3.30 & 3.30 & 3.30 \\ 3.30 & \omega_{i,1}^x & 3.30 & 3.30 \\ 3.30 & 3.30 & \omega_{i,2}^x & 3.30 \\ 3.30 & 3.30 & 3.30 & \omega_{i,2}^x \end{bmatrix} \cdot \begin{bmatrix} \Theta_{i,1} \\ \Theta_{i,2} \end{bmatrix} \cdot M_i \cdot \begin{bmatrix} \omega_{i,1} \cdot v_{i,1} \\ 3.10 \\ \omega_{i,2} \cdot v_{i,2} \\ 3.10 \end{bmatrix},
\]

where the notation “×” means skew-symmetric matrix. In Equations (6) all vectors and matrices are to be with respect to one and the same coordinate system. For elements with more than two nodes, for examples plates and shells, their elements are to be arranged in a similar order. The forces of elements in the common nodes are to be summarized.

The general Newton – Euler dynamic equations are applicable as for rigid, so for flexible systems and provide the basis for effective general recursive algorithms and program systems for multibody system dynamics analysis. The elements of the homogeneous transformation matrices are explicitly defined with respect to the deflections and computational expenses are
low. The numerical procedure uses constant relative mass and stiffness matrices for definition of the elastic and inertia forces. Here we do not discuss the methodology for application of the principle of virtual work for reduction of forces to the system coordinates and deriving the dynamic equations since that is well known procedure. Possible kinematic constraints are to be expressed in the same way as in the case of rigid bodies.

4 DYNAMICS OF STRUCTURES SUBJECT TO SEISMIC EXCITATIONS

The methodologies based on the FET have been successfully applied for large structures with many flexible elements and nodes. The methods of FERC and FEAC as they are presented above have common nature but are applicable for different kind of problems. Relative coordinates are successfully used for open branch multibody systems as single beam structures or systems compiled of many consequently connected adjacent elements as, robots and manipulators, large cantilever beam and corresponding structures, and etc. FEAC are used for complex structures with many adjacent flexible elements that form closed chains and possess many dependent coordinates. The approach could be successfully applied for simulation of complex structures subject to external disturbances like seismic excitations.

Two methods are used for simulation of seismic excitation, ground – structure force interaction method and the basement acceleration method. The first one uses data from simulation of ground structure interaction and the output data so obtained, the forces loading the basement, are input data for the dynamics analysis of the structure. This approach is the easiest for dynamics simulation of the structures and is fully compatible with the methods discussed above.

The acceleration based approach uses statistical data for the accelerograms for specific regions and the motion characteristics of the basement are input data for the dynamics simulation process. Actually, the basement acceleration and the integrated velocities and displacements are reonomic constraints for the dynamic equations. Further down an approach for solution of the dynamic equations subject to reonomic constraints and on this base application of the FERC and FEAC is presented.

Using the principle of the virtual work the forces are reduced to the system coordinates to derive the dynamic equations

\[ \mathbf{M} \cdot \ddot{\mathbf{q}} + \mathbf{B}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{S} \]  

(7)

where \( \mathbf{S} \) is \( n \times 1 \) matrix-vector of the generalized forces, \( \mathbf{B}(\mathbf{q}, \dot{\mathbf{q}}) \) is velocity depend term.

Earthquake shaking affects forced motion of the structure basement. For a region using statistical data the accelerograms for the space displacements are as follows:

\[ \ddot{q}_i = \ddot{q}_i(t), i = 1, 2, 3 \]  

(8)

These are reonomic kinematic constraints that depend on time. For a multibody system with dof \( n \) which motion is described by ODE, Equation (7), subject to \( m \) reonomic constraints, Equation (8), the dynamic equations are presented as follows:

\[ \mathbf{M} \begin{bmatrix} \ddot{q}_1 & \ddot{q}_{k+1} & \cdots & \ddot{q}_{m} \end{bmatrix} + \mathbf{B}(\mathbf{q}, \dot{\mathbf{q}}) = \begin{bmatrix} S_1 & \cdots & S_k & \cdots & S_{k+m} & \cdots & S_n \end{bmatrix} \]  

(9)

In Equation (9) the coordinates \( \ddot{q}_i = \ddot{q}_i(t), i = k + 1, k + 2, \ldots, k + m \) are known, while the generalized forces \( S_i = S_i(t), i = k + 1, k + 2, \ldots, k + m \) are unknown. The dynamic equations, Equation (9), are transformed as follows:

\[ \mathbf{M} \begin{bmatrix} \ddot{q}_1 & \ddot{q}_{k+1} & \cdots & \ddot{q}_n \end{bmatrix} + \mathbf{B}(\mathbf{q}, \dot{\mathbf{q}}) = -\mathbf{M} \begin{bmatrix} \ddot{q}_{k+1} & \cdots & \ddot{q}_{k+m} \end{bmatrix} + \mathbf{S} \]  

(10)

where the matrices \( \mathbf{S}, \mathbf{M}, \mathbf{M} \) are compiled according to the indices of the coordinates.
The famous beam of Kane et al. [4] is an example of the dynamic analysis subject to a reonomic constraint. It is a horizontal cantilever beam that rotates around a vertical axis. The rotation is prescribed by its acceleration and this example is a test for many researchers analyzing their methods. Here the same example is used to present the FERC and FEAC methods, as well as, the acceleration based approach for reonomic constraints. In Figure 4 the results of the simulation are presented. The motion characteristics of the beam input shaft (coordinate $q_1$) are presented in the Figure 4 (a). In Figures 4 (b, c) the beam topological graphs for FERC and FEAC are presented where in circles the dof of the nodes are shown. In Figures 4 (d, e) the results of the simulation process are depicted. The mass and inertia characteristics of the beam are: (all measures are in SI UNIT): modulus of elasticity $E = 7 \cdot 10^{10}$; shear modulus $G = 2.8 \cdot 10^{10}$; length $L = 10$.; density $\rho = 3000$; cross section area $S = 4 \cdot 10^{-4}$; cross-section moments of inertia $I_x = I_z = \frac{1}{2} I_c = 2.10^{-7}$. The beam discretization is of three elements. As it could be seen from the example using the FERC the results are compatible with that presented in [4], while using FEAC the deviations from the first example could be observed. That could be explained with the fact that for the FEAC method the coordinates presenting the global motions are not separated from these of the relative flexible deflections.

\begin{align*}
Z_0 & \triangleq q_1 \\
X_0 = X_b & = -1.06
\end{align*}

\begin{align*}
\dot{q}_1 &= \begin{cases} 
0.4|q - 7.5/\pi \sin(t/\pi/7.5)| & t < 15 \text{sec.} \\
6 \text{ rad/sec.} & t \geq 15 \text{ sec}
\end{cases} \Rightarrow \pi = 3.1416
\end{align*}

\begin{figure}[h]
\centering
(a) \hspace{1cm} (b) \hspace{1cm} (c)
\caption{Kane’s flexible beam [4]: (a) the beam and driving motion; (b) beam topological graph for FERC; (c) beam topological graph for FEAC; (d) results of the FERC; (e) results of the FEAC.}
\end{figure}

5 \ EXAMPLE OF A WIND POWER GENERATOR SUBJECT TO EARTHQUAKE

In Figure 5 the design scheme of a wind power generator is presented. The pillar is of three flexible elements, of which the gondola is set up. The electric power generator and its foundation (a rigid body) are considered free bodies since they are mounted on the gondola by bearings of which the stiffness and damping properties are taken into account. The rotating hub is also a rigid body connected to the gear shaft via a flexible coupling. The bearing of the hub shaft is rigidity set to the gondola. Each of the three flexible wings is discretized on three beam elements with changing size. The reducer ratio is 1/100 and the optimal angular velocity $\omega_w$ of the electric generator is 300 rad/sec. The seismic excitations are presented schematically either by forces caused by virtual springs and dampers ($k_{g_i}, c_{g_i}, i = 1,2$), or by the possible motion in plane (coordinates $q_1, q_2$). In the paper the acceleration method is used for presentation of the ground shaking for which the acceleration of the basement is $q_1 = 0.2G \pi \cos(\pi t)$. In Figure 6 (a) the rotor, as well as, the virtual spring and damper properties of the bearings that connect the
rotor to the gondola are depicted. The rotor is on two bearings $L_1, L_2$ of which the stiffness and damping properties are presented by virtual springs and dampers with coefficients $c_{b_i}, d_{b_i}, i = 1,2$. The following input data are used for the pillar – a beam of steel tube with height 20; density $\rho = 8.1 \times 10^3$; outer and inner diameters $D_0 = 2, D_1 = 1.96$, respectively; mass of the gondola (stator and reducer) $2.3 \times 10^3$; rotor mass $m_c = 1.6 \times 10^3$; mass inertia moment of the rotor $J_c = 100$. The rotor is with eccentricity $e = 0.002$ that causes dynamic vibrations. The values of the bearing coefficients are: $c_{b_i} = 9.6 \times 10^6, d_{b_i} = 10 \times 10^3, i = 1,2$. The data for the wings (Figure 6 (b)) consequently for each element in decreasing order of the size are as follows: lengths - 10; cross section area - 0.1, 0.09, 0.08; modulus of the elasticity - 2.1$\times 10^{11}$, modulus of the angular elasticity - 0.7$\times 10^{11}$; cross section inertia moments $I_y = I_z - 0.04, 0.035, 0.03$; polar inertia moment $I_c = I_x = I_y + I_z - 0.08, 0.07, 0.06$. The mass of the hub is 600. The

![Figure 5](image_url). (a) Design scheme of a wind power generator and; (b) its kinematic and dynamic model

clutch between the reducer and the hub allows small flexible deflections in three translational and three rotational directions with spring and damping coefficients $c_w = 5 \times 10^4$ and $d_w = 3 \times 10^3$, respectively. In Figure 6 by arrows the possible three dimensional flexible translations and rotational deformations of element nodes are depicted by arrows.

In Figure 7 (a) the results of dynamics simulation of the rotor deviations of the electric power generator for the operational time starting from zero angular velocity $\omega_0$ are shown. The transient nonlinear phenomena as a result of the rotor eccentricity are visible at angular velocity $\omega_e=24.04$ and $\omega_e=61.58$ are visible that is confirmed from the theoretical investigations presented in [21]. In Figure 6 (b) the results of the rotor deviations obtained during seismic excitation $q_1 = 0.2G\pi t \cos(\pi t)$ are presented. The excitation starts after the 5-th second of the operational time and lasts for 12 seconds. In Figure 8 (a – c) and 9 (a – c) the results of the structure dynamics simulation of the wing and pillar tips are also presented.

**Acknowledgement:** The authors acknowledge the financial support of Ministry of Education, Youth and Science, contract No. DUNC 01/3 – 2009.

**REFERENCES**


Figure 6. (a) The rotor and the pillar, and (b) The clutch, hub and one of the wings.

Figure 7. Rotor displacements: (a) operational time \( t = 80 \) sec. \( 0 \leq \omega \leq 120 \); (b) subject to seismic excitation within the time interval \( 5 \leq t \leq 12 \) s.

Figure 8. Deviation in X direction of the wings tip subject to seismic excitation (a) without rotation of the wings; (b) with rotation of the wings.

Figure 9. Deviation in Y direction of the wings tip subject to seismic excitation (a) without rotation of the wings; (b) with rotation of the wings.


Three-dimensional nonlinear shell theory for flexible multibody dynamics

S. L. Han*, O. A. Bauchau†

* University of Michigan-Shanghai Jiao Tong University Joint Institute
Shanghai, 200240, China
han1@sjtu.edu.cn

† Department of Mechanical and Aerospace Engineering
Hong Kong University of Science and Technology
Hong Kong, China
bauchau@ust.hk

1 ABSTRACT

In structural analysis, many components are approximated as plates. More often than not, classical plate theories, such as Kirchhoff or Reissner-Mindlin plate theories, form the basis of the analytical developments. The advantage of these approaches is that they lead to simple kinematic descriptions of the problem: the plate’s normal material line is assumed to remain straight and its displacement field is fully defined by three displacement and two rotation components. While such approach is capable of capturing the kinetic energy of the system accurately, it cannot represent the strain energy adequately. For instance, it is well known from three-dimensional elasticity theory that the normal material line will warp under load for laminated composite plates, leading to three-dimensional deformations that generate complex stress states. To overcome this problem, several high-order, refined plate theories have been proposed. While these approaches work well for some cases, they often lead to inefficient formulations because they introduce numerous additional variables. This paper presents a different approach to the problem: based on a finite element semi-discretization of the normal material line, plate equations are derived from three-dimensional elasticity using a rigorous dimensional reduction procedure.

Keywords: Plate theory, Structural analysis, Composite materials, Dimensional reduction

2 INTRODUCTION

Plates are structural components for which one dimension is far smaller than the other two. The mid-plane of the plate lies along its two long dimensions, and the normal to the plate extends along the shorter dimension. The material properties of the plate are assumed to vary smoothly over the plate’s mid-plane surface.

Numerous structures can be approximated as plates or shells. The long, slender wings of an aircraft can be analyzed, to a first approximation, as beams, but a more refined analysis will treat the upper and lower skins of the wing as thin plates or shells supported by ribs and longerons or stiffeners. The same can be said about helicopter or wind turbine blades. Buckling of the face sheets of wind turbine rotor blades is an important problem that cannot be captured by beam models. This instability, however, will be captured by plate models.

Solid mechanics theories describing plates, more commonly referred to as “plate theories,” play an important role in structural analysis because they provide tools for the analysis of these commonly used structural components. Although more sophisticated formulations, such as three-dimensional elasticity theory, could be used for the analysis of plates and shells, the associated computational burden is often too heavy. Plate theories reduce the analysis of complex, three-dimensional structures to two-dimensional problems. The main goal of the proposed plate theory is to approximate the three-dimensional plate-like structure with a two-dimensional model, while retaining an accurate representation of the local, three-dimensional stress and strain fields through the thickness of the plate.
Several plate theories have been developed based on various assumptions, and lead to different levels of accuracy. One of the simplest and most useful of these theories is due to Kirchhoff who analyzed the bending of thin plates. Kirchhoff plate theory [1, 2] is used commonly in many civil, mechanical and aerospace applications, although shear deformable plate theories [3, 4, 5], often called “Reissner-Mindlin plates,” have also found wide acceptance.

In many applications, however, plates are complex built-up structures. In aeronautical constructions, for instance, the increased use of laminated composite materials leads to heterogeneous, highly anisotropic structures. Layers of anisotropic material are stacked through the thickness of the plate. This new type of structural component prompted the development of new plate theories [6, 7, 8], often based on classical lamination theory [9, 10].

Further refinements then followed with the goal of capturing the intricate three-dimensional stress field that develops under load, with special emphasis on interlaminar shear stresses. The various approaches fall into two categories: higher-order and layer-wise plate theories. Higher-order plate theories have been proposed by assuming higher-order displacement fields for the entire normal material line [11, 12, 13, 14]. In layer-wise theories [15, 16, 17], the displacement field of the normal material line for each layer is independent of that of the others, with a simple $C_0$ continuity conditions imposed at the layer boundaries. Although these approaches lead to higher accuracy, the number of unknowns increases considerably. A comprehensive review of the literature is given by Noor and Burton [18].

Efficient plate models can be obtained more rigorously from three-dimensional elasticity through dimensional reduction techniques that split the original problem into a two-dimensional analysis over the plate’s mid-plane surface and a one-dimensional, through-the-thickness linear analysis. The plate’s stiffness matrix is a by-product of the dimensional reduction process, which also enables the recovery of three-dimensional stress fields. These approaches, derived from three-dimensional elasticity theory directly, can handle laminated plates and shells made of anisotropic composite materials without increasing the total number of unknowns.

Asymptotic and multiscale analysis methods have been the tools of choice for dimensional reduction. For plate problems, the thickness, denoted $h$, is typically far smaller than a representative span-wise dimension, denoted $a$, and the aspect ratio, $\varepsilon = h/a$, is a small parameter. Consequently, the gradients of stress and displacement components over the plate’s mid-surface are smaller than those through its thickness. Asymptotic and multiscale methods expand the solution in terms of the aspect ratio, leading to a rational decomposition of the three-dimensional problem into two-dimensional equations over the plate’s mid-plane and a through-the-thickness problem. Based on this approach, Friedrichs and Dressler [19] and Kalamkarov [20] investigated isotropic and in-homogenous plate problems, respectively.

In the Variational Asymptotic Method (VAM) proposed by Berdichevsky [21, 22], asymptotic analysis is applied to the energy functional. A unified theory based on VAM, presenting both linear, one-dimensional through-the-thickness analysis, and nonlinear, two-dimensional analysis over the plate or shell’s mid-plane surface was developed by Sutyrin and Hodges [23, 24], and Yu et al. [25, 26]. The plate’s $8 \times 8$ stiffness matrix relating the eight deformation measures to the corresponding eight stress resultants is a by-product of the linear, through-the-thickness analysis. Furthermore, the local strain field at all points through the plate’s thickness can be recovered once the global strain measures are known.

More recently, Kim [27] proposed a finite element based asymptotic analysis for generally anisotropic plate problems. The dimensional reduction process is based on a formal asymptotic expansion, which splits the three-dimensional plate problem into two sets of recursive equations: a set of one-dimensional local recursive problems and a set of two-dimensional global recursive problems.

It is not necessary to use asymptotic expansion methods to tackle plate analysis. Giavotto et al. [28] used a finite element based, semi-discretization approach to solve beam problems present-
ing arbitrary cross-sectional geometries and material properties. An adaptation of this approach to anisotropic, heterogeneous laminated composite plates was proposed by Masarati and Ghiringhelli [29], who found solutions of the three-dimensional equilibrium equations through the plate’s thickness. These solutions then yield the plate’s global compliance matrix and the local stress field can be recovered.

The Representative Volume Element (RVE) approach is a common tool for multiscale analysis. Recently, Gruttmann and Wagner [30] used a through-the-thickness RVE to develop local/global plate models. The displacement field of the RVE is split into rigid normal material line and warping components. The local RVE and global plate models are coupled at the lateral surfaces of the RVE and are solved simultaneously.

In this paper, a novel dimensional reduction technique for plate problems is proposed. A semi-discretization of the general equations of three-dimensional elasticity is performed, defining the “local model.” The equations relating the stress resultants, the plate’s deformation measures, and the warping field of the normal material line are derived from a linear combination of the equations of the local model. These equations define the relationship between the stress resultants and deformation measures in an implicit manner, and hence, define the “global model.” By means of a recursive process, power series solutions are found for the combined equations of the local and global models. Based on these solutions, the local problem for plate-like structures is reduced to the corresponding global problem, and the local stress and strain fields can recovered from the global solution. The 8×8 stiffness matrix of the plate is a by-product of the reduction process; it takes into account warping effects due to material heterogeneity. Local stress and strain fields at any point through the thickness of the plate can be recovered from the global solution. The proposed method is applicable to anisotropic plates with arbitrarily complex through-the-thickness lay-up configurations.

In a first step towards the development of a comprehensive framework for modeling the dynamics of plate and shells, a static solution of the problem is presented here. Because the solution is derived from three-dimensional, linear elasticity theory without the customary assumptions of plate or shell theories, it should provide a sound basis for generalization to the more challenging problems offered by dynamic systems.

3 KINEMATICS OF THE PROBLEM

Figure 1 depicts a flat plate of thickness \( h \) and a typical material line, denoted \( \mathcal{L} \), normal to the midplane of the plate. The volume of the plate is generated by sliding the normal material line over the plate’s midplane. Coordinate \( \alpha_1 \) and \( \alpha_2 \) define the plate’s midplane, i.e., they measure length along unit vectors \( \mathbf{b}_1 \) and \( \mathbf{b}_2 \), respectively. Point \( \mathbf{B} \) is located at the intersection of the normal material line with the plate’s midplane. The unit tangent vectors to midplane are defined as

\[
\mathbf{b}_1 = \frac{\partial \mathbf{x}_B}{\partial \alpha_1} \quad \text{and} \quad \mathbf{b}_2 = \frac{\partial \mathbf{x}_B}{\partial \alpha_2},
\]

where \( \mathbf{x}_B(\alpha_1, \alpha_2) \) is the position vector of point \( \mathbf{B} \) with respect to the origin of the inertial frame, \( \mathcal{F} = \{\mathbf{O}, \mathcal{I} = (\mathbf{i}_1, \mathbf{i}_2, \mathbf{i}_3)\} \). The normal material line is aligned with unit vector \( \mathbf{b}_3 \), which is normal to both \( \mathbf{b}_1 \) and \( \mathbf{b}_2 \), and material frame \( \mathcal{F}' = \{\mathbf{B}, \mathcal{I}' = (\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3)\} \) is then defined. A set of material coordinates that naturally represents the configuration of the plate is selected as \( \alpha_1, \alpha_2, \) and \( \alpha_3 \), where \( \alpha_3 \) is the coordinate through the plate’s thickness.

It will be convenient to develop the model using a non-dimensional formulation. A representative dimension of the plate’s thickness, denoted \( a_r \), will be used to normalize length and hence, \( \bar{\alpha}_i = \alpha_i/a_r, \ i = 1, 2, 3 \), are the non-dimensional coordinates of the problem and notation \( (\cdot)_j = a_r \frac{\partial (\cdot)}{\partial \alpha_j}, \ j = 1, 2 \), indicates a derivative with respect to non-dimensional span-wise variable \( \alpha_j \).

The rotation tensor that brings basis \( \mathcal{I} \) to basis \( \mathcal{I}' \) is denoted \( \mathbf{R} \) and the following motion tensor [31] is defined

\[
\mathbf{\mathcal{L}} = \begin{bmatrix} \mathbf{R} & \mathbf{r}_B \mathbf{R} \\ 0 & \mathbf{R} \end{bmatrix},
\]

(1)
Motion tensor \( \mathcal{C} \) represents a finite rigid-body motion from frame \( \mathcal{F} \) to \( \mathcal{F}^* \). The components of the plate’s generalized curvature tensors in its initial configuration, resolved in frame \( \mathcal{F}^* \), are then defined as

\[
\mathcal{K}_1^* = \mathcal{C}^{-1} \mathcal{K}_1, \quad \mathcal{K}_2^* = \mathcal{C}^{-1} \mathcal{K}_2, \tag{2a, 2b}
\]

where notation \((\cdot)^*\) indicates tensor components resolved in frame \( \mathcal{F}^* \). Because the plate is flat, rotation tensor \( \mathcal{R} \) is independent of coordinates \( \alpha_1 \) and \( \alpha_2 \), and hence,

\[
\mathcal{K}_1^* = \begin{bmatrix} 0 & \hat{b}_1^1 \\ 0 & 0 \end{bmatrix}, \quad \mathcal{K}_2^* = \begin{bmatrix} 0 & \hat{b}_2^2 \\ 0 & 0 \end{bmatrix}, \tag{3a, 3b}
\]

where notation \( \hat{b} \) indicates the skew symmetric matrix constructed from the components of vector \( \hat{b} \), i.e., \( \hat{b} = \text{axial}(\hat{b}) \).

### 3.1 Position and base vectors in the reference configuration

With these definitions, the position vector of an arbitrary material point of the plate becomes

\[
r(\alpha_1, \alpha_2, \alpha_3) = r_b(\alpha_1, \alpha_2) + \alpha_3 \hat{b}_3 = r_b(\alpha_1, \alpha_2) + q(\alpha_3), \tag{4}
\]

where vector \( q = r - r_b = \alpha_3 \hat{b}_3 \) defines the relative position of material point \( P \) with respect to point \( B \).

The base vectors associated with these material coordinates are \( g_{1} = \partial r / \partial \alpha_1 = \hat{b}_1 \), \( g_{2} = \partial r / \partial \alpha_2 = \hat{b}_2 \), and \( g_{3} = \partial r / \partial \alpha_3 = \hat{b}_3 \). The components of these base vectors resolved in basis \( \mathcal{B}^* \) are found as \( g_1^{*T} = \{1, 0, 0\} \), \( g_2^{*T} = \{0, 1, 0\} \), and \( g_3^{*T} = \{0, 0, 1\} \).

### 3.2 Position and base vectors in the deformed configuration

After deformation, the position vector of a material point becomes

\[
\mathcal{R}(\alpha_1, \alpha_2, \alpha_3) = r + \mathcal{U} = r + u_i \hat{b}_i, \tag{5}
\]

where \( \mathcal{U} \) is the displacement vector and its components resolved in basis \( \mathcal{B}^* \) are denoted \( u_i^* \), i.e., \( \mathcal{U} = u_1^* \hat{b}_1 + u_2^* \hat{b}_2 + u_3^* \hat{b}_3 \). The base vectors in the deformed configuration are now readily obtained...
\[
G_1 = \frac{\partial R}{\partial \alpha_1} = g_1 + \frac{\partial u^*}{\partial \alpha_1} b_1, \\
G_2 = \frac{\partial R}{\partial \alpha_2} = g_2 + \frac{\partial u^*}{\partial \alpha_2} b_1, \\
G_3 = \frac{\partial R}{\partial \alpha_3} = g_3 + \frac{\partial u^*}{\partial \alpha_3} b_1, \\
\]

Resolving these vectors in basis \( \mathcal{B}^* \) yields the components of the base vectors of the deformed configuration as

\[
G_i^* = \left\{ \begin{array}{c}
1 + \frac{\overline{u}_{1,1}}{\overline{u}_{2,1}} \\
\frac{\overline{u}_{1,2}}{\overline{u}_{2,2}} \\
\frac{\overline{u}_{1,3}}{1 + \overline{u}_{3,3}}
\end{array} \right\}, \quad G_j^* = \left\{ \begin{array}{c}
1 + \frac{\overline{u}_{1,2}}{\overline{u}_{3,2}} \\
\frac{\overline{u}_{2,2}}{1 + \overline{u}_{3,3}}
\end{array} \right\}, \quad G_k^* = \left\{ \begin{array}{c}
\frac{\overline{u}_{1,3}}{1 + \overline{u}_{3,3}}
\end{array} \right\},
\]

where the non-dimensional displacement vector is \( \overline{u} = u/a_r \). These expressions then yield the components of the deformation gradient tensor resolved in basis \( \mathcal{B}^* \)

\[
F^* = \begin{bmatrix}
1 + \overline{u}_{1,1} & \overline{u}_{1,2} & \overline{u}_{1,3} \\
\overline{u}_{2,1} & 1 + \overline{u}_{2,2} & \overline{u}_{2,3} \\
\overline{u}_{3,1} & \overline{u}_{3,2} & 1 + \overline{u}_{3,3}
\end{bmatrix}.
\]  

3.3 Strain components

It is now assumed that the plate undergoes small motions and that strains remain very small at all times. For this small-strain problem, the Green-Lagrange strain tensor reduces to \( \gamma^* = (E^* + E^{**})/2 - I \). For simplicity, the components of the strain tensor are split into the in-plane components, denoted \( \gamma'_i \), and the out-of-plane components, denoted \( \gamma'_o \). Explicit expressions of the strain components are then obtained from eq. (8) as

\[
\gamma'_i = \left\{ \begin{array}{c}
\gamma'_{i1} \\
\gamma'_{i2} \end{array} \right\} = \left\{ \begin{array}{c}
\frac{\overline{u}_{1,1}}{\overline{u}_{1,2} + \overline{u}_{2,1}} \\
\frac{\overline{u}_{2,2}}{\overline{u}_{1,2} + \overline{u}_{2,1}}
\end{array} \right\} = \overline{A}^i_1 \overline{u}_1 + \overline{A}^i_2 \overline{u}_2,
\]

\[
\gamma'_o = \left\{ \begin{array}{c}
\gamma'_{o1} \\
\gamma'_{o2} \end{array} \right\} = \left\{ \begin{array}{c}
\frac{\overline{u}_{1,3}}{\overline{u}_{2,3} + \overline{u}_{3,2}} \\
\frac{\overline{u}_{2,3}}{\overline{u}_{2,3} + \overline{u}_{3,2}}
\end{array} \right\} = \overline{A}^o_1 \overline{u}_1 + \overline{A}^o_2 \overline{u}_2 + B^O \overline{u},
\]

respectively. The components of Green-Lagrange strain tensor are collected into a single array,

\[
\gamma^* = \left\{ \begin{array}{c}
\gamma'_i \\
\gamma'_o
\end{array} \right\} = A_{11} \overline{u}_1 + A_{12} \overline{u}_2 + B \overline{u}.
\]

In eqs. (9), the following differential operators were defined

\[
A^i_1 = \begin{bmatrix}
1 & 0 & 0 \\
0 & 0 & 0 \\
0 & 1 & 0
\end{bmatrix}, \quad A^i_2 = \begin{bmatrix}
0 & 0 & 0 \\
0 & 1 & 0 \\
1 & 0 & 0
\end{bmatrix},
\]

and

\[
A^o_1 = \begin{bmatrix}
0 & 0 & 0 \\
0 & 0 & 1 \\
0 & 0 & 0
\end{bmatrix}, \quad A^o_2 = \begin{bmatrix}
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 1 & 0
\end{bmatrix}, \quad B^O = \begin{bmatrix}
0 & 0 & 1 \\
1 & 0 & 0 \\
0 & 1 & 0
\end{bmatrix} \frac{\partial}{\partial \alpha_3}.
\]

Finally, in eq. (10), the following differential operator was defined

\[
\overline{A}^i_1 = \begin{bmatrix}
\overline{A}^{*i}_1 \\
\overline{A}^{*i}_2
\end{bmatrix}, \quad \overline{A}^i_2 = \begin{bmatrix}
\overline{A}^{*i}_1 \\
\overline{A}^{*i}_2
\end{bmatrix}, \quad \overline{B} = \begin{bmatrix}
\overline{B}^i \\
\overline{B}^o
\end{bmatrix}. 
\]

447
3.4 Semi-discretization

Plate theory is characterized by two-dimensional differential equations governing the displacement field assumed to be a function of the in-plane variables, \( \alpha_1 \) and \( \alpha_2 \), only. In the above paragraphs, the displacement field has been treated as a general vector field depending on three independent variables, \( \alpha_1 \), \( \alpha_2 \), and \( \alpha_3 \). To obtain a two-dimensional formulation, the following semi-discretization of the displacement field is performed,

\[
\hat{u}(\alpha_1, \alpha_2, \alpha_3) = N(\alpha_3) \hat{u}(\alpha_1, \alpha_2),
\]

where matrix \( N(\alpha_3) \) stores the one-dimensional shape functions used in the discretization and array \( \hat{u}(\alpha_1, \alpha_2) \) stores the nodal values of the displacement field. Let \( N \) be the number of nodes used to discretize the plate’s normal material line; \( 3N \) is then the total number of degrees of freedom.

Equation (14) corresponds to a semi-discretization of the problem, as first proposed by Giavotto et al. [28] for beams, and Masarati and Ghiringhelli [29] for plates. Figure 2 depicts the finite element mesh that extends over the plate’s normal material line only and the nodal values of the displacement components remain functions of the in-plane variables, \( \alpha_1 \) and \( \alpha_2 \) (For clarity, stresses are shown on one face of the differential element only). Consequently, the shape functions depend on the variables describing the geometry of the normal material line, \( \alpha_3 \), only, i.e., \( N = N(\alpha_3) \). The dependency of the nodal displacement components on the in-plane variables is explicitly stated as \( \hat{u} = \hat{u}(\alpha_1, \alpha_2) \). The semi-discretization procedure leads to a numerical treatment of the solution for variable \( \alpha_3 \) whereas the dependency of the solution on variables \( \alpha_1 \) and \( \alpha_2 \) is treated analytically.

3.5 Strain components

Introducing this discretization into eq. (10) yields the components of Green-Lagrange strain tensor as

\[
\gamma^a = A_1 N \hat{u}_1 + A_2 N \hat{u}_2 + B N \hat{u}.
\]

After the semi-discretization, the strains components depend on the following displacement derivatives only, \( \gamma^a = \{u_1^a, u_2^a, u_1^a + u_2^a, u_1^a, u_1^a, u_2^a, u_2^a, u_1^a + u_2^a, u_1^a + u_2^a, u_1^a + u_2^a\} \), which can be expressed as \( \gamma^a = \tilde{a}_1 \tilde{u}_1 + \tilde{a}_2 \tilde{u}_2 \), where Boolean matrices \( \tilde{a}_1 \) and \( \tilde{a}_2 \) are defined by eq. (70). Equation (15) now reduces to

\[
\gamma^a = \tilde{a}_1 N \hat{u} + \tilde{a}_2 N \hat{u},
\]

where operator \( \tilde{B} \) is defined by eq. (13) and operator \( \tilde{A} = \tilde{A}_1 \tilde{b}_1 + \tilde{A}_2 \tilde{b}_2 \), where Boolean matrices \( \tilde{b}_1 \) and \( \tilde{b}_2 \) are defined by eq. (71). Note that \( \tilde{A}_1 = A_1 \tilde{a}_1^T \) and \( \tilde{A}_2 = A_2 \tilde{a}_2^T \).
3.6 Rigid-body motions

For convenience, an infinitely small rigid-body motion is introduced, \( \mathcal{X}_R = \{ \mathbf{u}_R^T, \phi_R^T \} \), where \( \mathbf{u}_R \) are the components of a rigid-body translation and \( \phi_R \) those of an infinitesimal rigid-body rotation. Using the motion tensor defined by eq. (1), the components of the rigid-body motion, resolved in material frame, are found as \( \mathcal{X}_R = \mathbf{C}^{-1} \mathcal{X}_R \). Because array \( \mathcal{X}_R \) represents a rigid-body motion, its spatial derivative vanishes, \( \mathcal{X}_R = 0 \), and hence, \( \mathcal{X}_R^* = -\mathbf{C}^{-1} \mathcal{X}_R \), where the second equality results from the definition of generalized curvature tensor in eq. (2a). A similar reasoning yields \( \mathcal{X}_{R2}^* = -\mathbf{C}_2^{-1} \mathcal{X}_R^* \).

An infinitely small rigid-body displacement field of the plate can be expressed as

\[
\begin{bmatrix}
\mathbf{u}_1 \\
\mathbf{u}_2 \\
\mathbf{u}_3
\end{bmatrix} =
\begin{bmatrix}
1 & 0 & 0 & 0 & \alpha_3 & 0 \\
0 & 1 & 0 & -\alpha_3 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
\mathbf{u}_R^* \\
\phi_R^*
\end{bmatrix} = \bar{\mathbf{Z}} \mathcal{X}_R^* = \mathcal{N} \mathbf{Z} \mathcal{X}_R^*,
\]

(17)

where matrix \( \bar{\mathbf{Z}} \) stacks the rows of matrix \( \mathbf{Z} \) for each of the nodes of the model. Note that the last component of the rotation vector corresponds to a rotation of the normal material line about unit vector \( \mathbf{b}_3 \). This rotation component is, of course, immaterial, resulting in the vanishing of the last column of matrix \( \bar{\mathbf{Z}} \) and the last row and column of the generalized curvature tensors, \( \mathcal{X}_1^* \) and \( \mathcal{X}_2^* \).

The nodal components of the rigid-body displacement field now become \( \mathbf{u} = \bar{\mathbf{Z}} \mathcal{X}_R^* \), and their derivatives are \( \mathbf{u}_1 = -\mathcal{X}_1^* \mathbf{Z} \mathcal{X}_R^* \) and \( \mathbf{u}_2 = -\mathcal{X}_2^* \mathbf{Z} \mathcal{X}_R^* \). Introducing these results into eq. (15) leads to \( \mathcal{X}_R^* = -(\mathbf{A}_N) \mathbf{Z} \mathcal{X}_1^* -(\mathbf{A}_N) \mathbf{Z} \mathcal{X}_2^* + (\mathbf{B}_N) \mathbf{Z} \mathcal{X}_R^* \) and because the strain field associated with an arbitrary rigid-body motion vanishes, it follows that

\[
(\mathbf{A}_N) \mathbf{Z} \mathcal{X}_1^* -(\mathbf{A}_N) \mathbf{Z} \mathcal{X}_2^* + (\mathbf{B}_N) \mathbf{Z} \mathcal{X}_R^* = 0
\]

(18)

3.7 Nodal forces

As was done for the components of the Green-Lagrange strain tensor in eq. (10), the components of stress tensor are split into their in-plane and out-of-plane components, denoted \( \mathcal{X}^* = \{ \mathcal{X}_1^*, \mathcal{X}_2^* \} \) and \( \mathcal{X}_0 = \{ \mathcal{X}_3^*, \mathcal{X}_4^*, \mathcal{X}_5^* \} \), respectively. The array of stress components then becomes \( \mathcal{X}^* = \{ \mathcal{X}_{i}^T \mathcal{X}_{j}^T \} \).

If the plate is made of a linearly elastic material, a linear relationship holds between the components of the stress tensor and those of the strain components

\[
\mathcal{X}^* = \mathbf{Z}^* \mathcal{Y}^*,
\]

(19)

where the components of the \( 6 \times 6 \) material stiffness matrix resolved the material basis are denoted \( \mathbf{Z}^* \). The non-dimensional form of the constitutive laws is \( \mathcal{X}^* = \mathbf{Z}^* \mathcal{Y}^* \), where \( \mathcal{X}^* = \mathcal{X}^*/E_r, \mathbf{Z}^* = \mathbf{Z}^*/E_r, \) and \( E_r \) is a representative value of Young’s modulus.

Figure 2 shows a differential element of the plate subjected to stresses applied over its four vertical faces. Stress vectors \( \mathcal{X}_1^* \) and \( \mathcal{X}_2^* \) are acting on the faces normal to unit vectors \( \mathbf{b}_1 \) and \( \mathbf{b}_2 \), respectively, but are not independent. Indeed, let array \( \mathcal{X}^* = \{ \mathbf{x}_{11}, \mathbf{x}_{13}, \mathbf{x}_{22}, \mathbf{x}_{23}, \mathbf{x}_{12} \} \) store the independent stress components; it then follows that \( \mathbf{x}_{11} = \mathbf{a}_1 \mathbf{x}^* \) and \( \mathbf{x}_{22} = \mathbf{a}_2 \mathbf{x}^* \), where Boolean matrices \( \mathbf{a}_1 \) and \( \mathbf{a}_2 \) are defined by eq. (70). Introducing matrices \( \mathbf{a}^\top \mathbf{b} \) defined by eq. (72) then leads to

\[
\begin{bmatrix}
\mathbf{x}_{11} \\
\mathbf{x}_{22}
\end{bmatrix} = \mathbf{a}^\top \mathcal{X}^*, \quad \mathcal{X}^* = \mathbf{b} \begin{bmatrix}
\mathbf{x}_{11} \\
\mathbf{x}_{22}
\end{bmatrix}.
\]

(20)

The arrays of non-dimensional nodal forces acting on faces normal to unit vectors \( \mathbf{i}_1 \) and \( \mathbf{i}_2 \) are defined as \( \mathbf{P}_1 = \int_{\mathcal{X}} N^T \mathcal{X}_1 \, d\mathbf{a}_3 \), and \( \mathbf{P}_2 = \int_{\mathcal{X}} N^T \mathcal{X}_2 \, d\mathbf{a}_3 \), respectively. Introducing the discretized...
components of Green-Lagrange’s strain tensor given by eq. (15) and material constitutive laws (19) into the nodal forces yields

\[
\begin{align*}
\mathbf{\hat{P}}_1 &= \mathbf{M}_{11} \mathbf{\hat{u}}_1 + \mathbf{M}_{12} \mathbf{\hat{u}}_2 + \mathbf{C}^T \mathbf{\hat{u}}, \\
\mathbf{\hat{P}}_2 &= \mathbf{M}_{12}^T \mathbf{\hat{u}}_1 + \mathbf{M}_{22} \mathbf{\hat{u}}_2 + \mathbf{C}^T \mathbf{\hat{u}},
\end{align*}
\]

(21a)

(21b)

where the following matrices, all of size \(3N \times 3N\), were defined

\[
\begin{align*}
\mathbf{\bar{M}}_{11} &= \int_{\mathcal{E}} (\bar{A} N)^T \bar{Z}^* (\bar{A} N) \, d\alpha_3, \\
\mathbf{\bar{M}}_{12} &= \int_{\mathcal{E}} (\bar{A} N)^T \bar{Z}^* (\bar{A} \bar{N}) \, d\alpha_3, \\
\mathbf{\bar{M}}_{22} &= \int_{\mathcal{E}} (\bar{A} \bar{N})^T \bar{Z}^* (\bar{A} N) \, d\alpha_3, \\
\mathbf{\bar{C}}_1 &= \int_{\mathcal{E}} (\bar{B} N)^T \bar{Z}^* (\bar{A} N) \, d\alpha_3, \\
\mathbf{\bar{C}}_2 &= \int_{\mathcal{E}} (\bar{B} \bar{N})^T \bar{Z}^* (\bar{A} \bar{N}) \, d\alpha_3.
\end{align*}
\]

(22a)

(22b)

(22c)

Given the distribution of material stiffness properties, these matrices can be evaluated by integration through the thickness of the plate.

Because stress vectors \(\mathbf{\xi}_1\) and \(\mathbf{\xi}_2\) are not independent, the same is true of the corresponding nodal forces. Equation (20) implies

\[
\begin{bmatrix}
\mathbf{\hat{P}}_1 \\
\mathbf{\hat{P}}_2
\end{bmatrix} = \hat{\mathbf{\Theta}} \mathbf{\hat{P}} = \hat{\mathbf{\Theta}} \begin{bmatrix}
\mathbf{\hat{P}}_1 \\
\mathbf{\hat{P}}_2
\end{bmatrix}
\]

(23)

where array \(\mathbf{\hat{P}} = \int_{\mathcal{E}} N^T \bar{Z} \, d\alpha_3\) stores the arrays of independent forces at all nodes. The explicit expression of array \(\mathbf{\hat{P}}\) is

\[
\mathbf{\hat{P}} = \mathbf{\bar{M}} \mathbf{\hat{u}} + \mathbf{\bar{C}} \mathbf{\hat{u}}.
\]

(24)

where matrices \(\mathbf{\bar{M}}\) and \(\mathbf{\bar{C}}\), both of size \(5N \times 5N\), are

\[
\begin{align*}
\mathbf{\bar{M}} &= \int_{\mathcal{E}} (\bar{A} N)^T \bar{Z}^* (\bar{A} N) \, d\alpha_3, \\
\mathbf{\bar{C}} &= \int_{\mathcal{E}} (\bar{B} N)^T \bar{Z}^* (\bar{A} N) \, d\alpha_3.
\end{align*}
\]

(25a)

(25b)

Because \(\bar{A}_1 = \bar{A} \bar{a}_1^T\) and \(\bar{A}_2 = \bar{A} \bar{a}_2^T\), the following matrix identities follow

\[
\begin{bmatrix}
\mathbf{\bar{M}}_{11} & \mathbf{\bar{M}}_{12} \\
\mathbf{\bar{M}}_{12}^T & \mathbf{\bar{M}}_{22}
\end{bmatrix} = \hat{\mathbf{\Theta}} \begin{bmatrix}
\bar{A} \bar{a}_1^T \bar{a}_1 \bar{a}_1^T \\
\bar{A} \bar{a}_2^T \bar{a}_2 \bar{a}_2^T
\end{bmatrix} = \hat{\mathbf{\Theta}} \bar{A} \bar{a}_1 \bar{a}_1^T
\]

(26)

\[
\begin{bmatrix}
\mathbf{\bar{C}}_1 & \mathbf{\bar{C}}_2 \\
\mathbf{\bar{C}}_2^T & \mathbf{\bar{C}}_1^T
\end{bmatrix} = \hat{\mathbf{\Theta}} \bar{C} \bar{a}_1 \bar{a}_1^T.
\]

(27)

3.8 Stress resultants

The sum of forces acting along the normal material line on the face normal to unit vector \(\bar{b}_1\) is \(\bar{F}_1 = \int_{\mathcal{E}} \bar{N}_1 \, d\alpha_3\), and the sum of the moments is \(\bar{M}_1 = \int_{\mathcal{E}} \bar{Q}_1 \, d\alpha_3\). Similar observations hold for the stress resultants acting on a face normal to unit vector \(\bar{b}_2\). Introducing the discretized components of Green-Lagrange’s strain tensor given by eq. (15), the nodal forces defined by eq. (21), and matrix \(\mathbf{Z}\) defined by eq. (17), the stress resultants become

\[
\mathbf{\bar{F}}_1 = \begin{bmatrix} \bar{F}_1 \\ \bar{M}_1 \end{bmatrix} = \bar{Z}^T \mathbf{\hat{P}}_1, \quad \mathbf{\bar{F}}_2 = \begin{bmatrix} \bar{F}_2 \\ \bar{M}_2 \end{bmatrix} = \bar{Z}^T \mathbf{\hat{P}}_2,
\]

(28)

where \(\mathbf{\bar{F}}_1\) and \(\mathbf{\bar{F}}_2\) are the array of stress resultants on faces normal to unit vectors \(\bar{b}_1\) and \(\bar{b}_2\), respectively, resolved in the material basis.

Stress resultants \(\mathbf{\bar{F}}_1\) and \(\mathbf{\bar{F}}_2\) are not independent but consist both of a subset of the eight independent stress resultant stored in the array \(\mathbf{\bar{F}}^{\mathcal{E}} = \{\bar{N}_1, \bar{N}_2, \bar{N}_{12}, \bar{Q}_1, \bar{Q}_2, \bar{M}_1, \bar{M}_2, \bar{M}_{12}\}\), where \(\bar{N}_1, \bar{N}_2, \bar{N}_{12}, \bar{Q}_1, \bar{Q}_2, \bar{M}_1, \bar{M}_2, \bar{M}_{12}\)

and $N_{12}$ are the in-plane forces, $Q_1$ and $Q_2$ the transverse shear forces, and $M'_1$, $M'_2$, and $M'_{12}$ the bending moments. It then follows that

$$\begin{aligned}
\left\{ \begin{array}{c}
\mathcal{F}_1 \\
\mathcal{F}_2
\end{array} \right\} = \left[ \begin{array}{cc}
\mathbf{a}_1 & 0 \\
0 & \mathbf{a}_2
\end{array} \right] \mathcal{F}^* = \mathbf{a} \mathcal{F}^*,
\mathcal{F}^* = \left[ \begin{array}{c}
\mathbf{q}_1 \\
\mathbf{q}_2
\end{array} \right] \mathcal{F}_1^* = \mathbf{q} \mathcal{F}_1^*,
\end{aligned}$$

(29)

where matrices $\mathbf{a}_1$, $\mathbf{a}_2$, $\mathbf{q}_1$, and $\mathbf{q}_2$ and $\mathbf{a}$ and $\mathbf{q}$ are defined by eqs. (73), (74), and (75), respectively. Finally, the relationship between the independent stress resultants and the corresponding nodal forces is written as

$$\mathcal{F}^* = \mathbf{G}^T \mathbf{P} = \mathbf{G}^T \left( \mathbf{M} \dot{\mathbf{y}} + \mathbf{C} \ddot{\mathbf{u}} \right).$$

(30)

where matrix $\mathbf{G}$ is defined as

$$\mathbf{G} = \mathbf{a}^T \mathbf{Z}^T \mathbf{a} + \hat{\mathbf{a}}^T \mathbf{Z} \hat{\mathbf{a}}.$$

(31)

The following matrix identities are verified easily

$$\begin{aligned}
\mathbf{a}^T \mathbf{Z} = \mathbf{G} \mathbf{b}_1, \\
\hat{\mathbf{a}}^T \mathbf{Z} = \mathbf{G} \mathbf{b}_2.
\end{aligned}$$

(32a)

$$\begin{aligned}
\mathbf{q}^T \mathbf{Z} = \mathbf{G} \mathbf{d}_1, \\
\hat{\mathbf{q}}^T \mathbf{Z} = \mathbf{G} \mathbf{d}_2.
\end{aligned}$$

(32b)

With these definitions, eq. (18) is recast in the following compact form

$$\left( \mathbf{A} \mathbf{N} \right) \mathbf{\tilde{K}} \mathbf{\tilde{x}}^* = \left( \mathbf{B} \mathbf{N} \right) \mathbf{Z} = 0.$$

(33)

where the combined curvature tensor is defined as

$$\mathbf{\tilde{x}}^* = \mathbf{a}_1^T \mathbf{x}_1^* + \mathbf{q}_2^T \mathbf{x}_2^*.$$

(34)

4 GOVERNING EQUATIONS

The plate’s governing equations will be derived from the principle of virtual work, which states that $\delta A - \delta W_E = 0$, where $\delta A$ and $\delta W_E$ denote the variation of strain energy and virtual work done by the external forces, respectively. These quantities will be evaluated in sections 4.1 and 4.2, respectively.

4.1 The strain energy density

Using the discretized components of Green-Lagrange’s strain tensor given by eq. (15), the strain energy per unit area of the plate, defined as $a = 1/2 \int_{\Omega} \varepsilon^T \varepsilon \mathbf{N} \mathbf{N} \mathbf{d} \alpha_{\Omega}$, becomes

$$\begin{aligned}
\bar{a} = \frac{1}{2} \left[ \hat{\mathbf{a}}^T \hat{\mathbf{P}} - \mathbf{a}^T \mathbf{P} + \hat{\mathbf{a}}^T \left( \mathbf{C} \hat{\mathbf{u}} + \mathbf{E} \ddot{\mathbf{u}} \right) \right],
\end{aligned}$$

(35)

where matrix, $\mathbf{E}$, of size $3N \times 3N$, is defined as

$$\mathbf{E} = \int_{\Omega} \left( \mathbf{B} \mathbf{N} \right)^T \mathbf{\tilde{K}}^* \left( \mathbf{B} \mathbf{N} \right) \mathbf{d} \alpha_{\Omega}.$$

(36)

For the rigid-body motions defined in section 3.6, the nodal forces, $\mathbf{P}_1$ and $\mathbf{P}_2$, and the strain energy density, $\bar{a}$, must vanish, leading to $\mathbf{M}_1 \ddot{\mathbf{u}}_1 + \mathbf{M}_2 \ddot{\mathbf{u}}_2 + \mathbf{C} \dot{\mathbf{u}} = 0$, $\mathbf{M}_1 \ddot{\mathbf{u}}_1 + \mathbf{M}_2 \ddot{\mathbf{u}}_2 + \mathbf{C} \dot{\mathbf{u}} = 0$, and $\mathbf{C} \ddot{\mathbf{u}}_1 + \mathbf{C} \ddot{\mathbf{u}}_2 + \mathbf{E} \ddot{\mathbf{u}} = 0$. When expressed in terms of the rigid-body motion components, these relationships become $(-\mathbf{M}_1 \ddot{\mathbf{x}}^*_1 - \mathbf{M}_2 \ddot{\mathbf{x}}^*_2 + \mathbf{C} \dot{\mathbf{x}}^*_1 \mathbf{y}^*_1 + \mathbf{E} \ddot{\mathbf{x}}^*_1 \mathbf{y}^*_1 = 0)$ and $(-\mathbf{M}_2 \ddot{\mathbf{x}}^*_1 - \mathbf{M}_1 \ddot{\mathbf{x}}^*_2 + \mathbf{C} \dot{\mathbf{x}}^*_2 \mathbf{y}^*_2 + \mathbf{E} \ddot{\mathbf{x}}^*_2 \mathbf{y}^*_2 = 0)$, respectively. Because the rigid-body motion is arbitrary, the following matrix identities result

$$\begin{aligned}
\mathbf{M}_1 \ddot{\mathbf{x}}^*_1 + \mathbf{M}_2 \ddot{\mathbf{x}}^*_2 - \mathbf{C} \dot{\mathbf{x}}^*_1 \mathbf{y}^*_1 = 0, \\
\mathbf{M}_1 \ddot{\mathbf{x}}^*_1 - \mathbf{M}_2 \ddot{\mathbf{x}}^*_2 - \mathbf{C} \dot{\mathbf{x}}^*_2 \mathbf{y}^*_2 = 0, \\
\mathbf{C} \ddot{\mathbf{x}}^*_1 \mathbf{y}^*_1 + \mathbf{C} \ddot{\mathbf{x}}^*_2 \mathbf{y}^*_2 - \mathbf{E} \ddot{\mathbf{x}}^*_1 = 0.
\end{aligned}$$

(37a)

(37b)

(37c)
Using eqs. (26), (27), and (32), these matrix identities reduce to

\[ C^T \bar{Z} = M \bar{G} \bar{X}^*, \]  
\[ E \bar{Z} = C G \bar{X}^*. \]  

(38a)  
\( (38b) \)

4.2 The virtual work done by the externally applied loads

As shown in fig. 2, the plate is subjected to three types of loading: surface tractions, \( p_L \) and \( p_H \), applied at the plate’s top and bottom surfaces, respectively, and body forces, \( h \). The virtual work done by these externally applied loads is then

\[ \delta W_E = \left[ \delta u^T p^* + \delta u_h^T p_h^* + \int_\Omega \delta u^T h^* \, d\alpha_3 \right] \, d\alpha_1 \, d\alpha_2 = \delta \bar{u}^T \bar{Q} \, d\alpha_1 \, d\alpha_2. \]  

(39)

The loading vector, \( \bar{Q} \), is found by introducing the discretized displacement field given by eq. (14) in the second equality to find \( \bar{Q} = N^T (h/2) p^*_h + N^T (-h/2) p^*_L + \int_\Omega N^T h^* \, d\alpha_3 \) and its non-dimensional counterpart, \( \bar{Q} \), is \( \bar{Q} = Q / (a_t E_t) \).

4.3 Equilibrium equations

Both global and local equilibrium equations will be obtained from the principle of virtual work. First, a virtual rigid-body motion is considered and because the strain energy vanishes for such motion, the principle reduces to \( \delta W_E = 0 \), or

\[ \delta \bar{W}_E = \delta \bar{\omega}^T \left[ -\bar{\lambda}_1^T \bar{\nu}_1 + \bar{\nu}_2^T \bar{\nu}_2 + \bar{\nu}_1 + \bar{\nu}_2 + \bar{Z} \bar{Q} \right] \, d\alpha_1 \, d\alpha_2 = 0. \]  

(40)

Because this expression must vanish for all arbitrary virtual rigid-body displacements, the bracketed term must vanish, yielding the plate’s global equilibrium equations,

\[ \bar{\nu}_1 - \bar{\lambda}_1^T \bar{\nu}_1 + \bar{\nu}_2^T \bar{\nu}_2 + \bar{Z} \bar{Q} = -\bar{\alpha}, \]  

(41)

where \( \bar{\alpha} = \bar{Z} \bar{Q} \) is the resultant of the externally applied loads.

Next, an arbitrary virtual displacement field is considered and the principle of virtual work yields the local equilibrium equations of the problem as

\[ \hat{P}_{1,1} + \hat{P}_{2,2} - C_{1,1} \bar{u}_1 - C_{1,2} \bar{u}_2 - E \bar{w} = -\bar{Q}, \]  

(42)

where integration by parts was used.

4.4 Global constitutive laws

The nodal displacement field is now decomposed as \( \bar{u} = \bar{Z} \bar{\omega}^* + \bar{\omega} \), where component \( \bar{Z} \bar{\omega}^* \) represents a rigid normal-material-line motion and component \( \bar{\omega} \) an additional warping field. Because expression “rigid normal-material-line motion” is wordy, it will be abbreviated as “rigid-normal motion” in the sequel. Note that the rigid-normal motion is not a rigid-body motion of the entire plate, and hence, generates strains. The proposed decomposition is arbitrary: indeed, an arbitrary rigid-normal motion can be selected and the corresponding warping field is then \( \bar{w} = \bar{u} - \bar{Z} \bar{\omega}^* \). Furthermore, this decomposition is redundant because the arbitrary warping field also contains a rigid-normal motion.

Equations (30) now yield the stress resultants, \( \bar{\lambda}^* = \bar{C} \bar{M} (a^T_0 (\bar{Z} \bar{\omega}^* + \bar{\omega}_1) + a^T_0 (\bar{Z} \bar{\omega}^* + \bar{\omega}_2)) + \bar{C}^T (\bar{Z} \bar{\omega}^* + \bar{\omega}) \) and identities (38a) and (31) then lead to

\[ \bar{\lambda}^* = (\bar{C}^T \bar{M} \bar{G}) \bar{\omega}^* + (\bar{M} \bar{G})^T (a^T_0 \bar{\omega}_1 + a^T_0 \bar{\omega}_2) + (\bar{C} \bar{G})^T \bar{\omega}. \]  

(43)
where the global strain components were defined as
\[ \varepsilon^g = \frac{1}{2} \mathcal{Z}^1 \varepsilon^1 + \frac{1}{2} \mathcal{Z}^2 \varepsilon^2 + \mathcal{Z}^3 \varepsilon^3. \]  
(44)

Note that this expression defines the global strain components in terms of the rigid-normal motion, \( \mathcal{Z}^k \), which is as yet undefined. In the absence of warping, \( i.e. \), when \( \mathbf{w} = 0 \), eq. (43) reduces to
\[ \mathcal{Z}^k = \mathcal{Z}^k_{RM} \varepsilon^k, \]
where \( \mathcal{Z}^k_{RM} = \mathcal{G}^T M G \) is the Reissner-Mindlin stiffness matrix. Indeed, Reissner-Mindlin plate theory postulates that normal material lines remain straight, implying that the sectional displacement field is captured adequately by the rigid-normal motion described by matrix \( \mathcal{Z}^k \).

The last two terms of eq. (43) describe the change in the sectional stiffness resulting from warping deformation.

In eq. (43), if the warping displacements, \( \mathbf{w} \), and their derivatives, \( \mathbf{w}_1 \) and \( \mathbf{w}_2 \), can be expressed in terms of the stress resultants, an equivalent global stiffness matrix can be found. Clearly, the determination of the warping field is key to the accurate evaluation of the plate’s global stiffness matrix.

### 4.5 Governing equations

Introducing the decomposition of the displacement field into eq. (42) leads to
\[ \mathbf{E} \mathbf{w} = \mathbf{A}_1 \mathbf{P}_1 + \mathbf{A}_2 \mathbf{P}_2 - \mathbf{C} \mathcal{E}^\kappa \mathbf{E}^\kappa - \mathbf{C}_1 \mathbf{w}_1 - \mathbf{C}_2 \mathbf{w}_2 + \mathbf{Q}. \]

Introducing the nodal forces from eq. (21) and combining the results with eq. (43) yields
\[ \mathbf{M}_1 \mathbf{\ddot{u}}_{11} + \mathbf{M}_{12} \mathbf{\ddot{u}}_{12} + \mathbf{M}_{22} \mathbf{\ddot{u}}_{22} + \mathbf{P}_1 \mathbf{\ddot{u}}_{11} + \mathbf{P}_2 \mathbf{\ddot{u}}_{22} - \mathbf{E} \mathbf{F} = - \mathbf{Q}, \]
(45)

where the following arrays were defined
\[ \mathbf{\ddot{u}} = \begin{bmatrix} \mathbf{\ddot{u}}^1 \\ \mathbf{\ddot{u}}^2 \end{bmatrix}, \quad \mathbf{Q} = \begin{bmatrix} \mathbf{Q}^1 \\ \mathbf{Q}^2 \end{bmatrix}. \]
(46)

Linear system (45) is a hybrid system that combines the local and global equilibrium equations. The local problem consists of the equilibrium equations of three-dimensional elasticity written in terms of nodal displacements whereas the global problem consists of the plate equations written in terms of stress resultants and global strain measures. These combined equations link the local and global problems in a formal manner.

Linear system (45) features \( N_T = 3N + 8 \) unknowns, the \( 3N \) nodal displacements and the \( 8 \) global strain measures. The following matrices, each of size \( N_T \times N_T \), have been defined
\[ \mathbf{\ddot{u}}_{11} = \begin{bmatrix} \mathbf{M}_{11} & 0 \\ 0 & 0 \end{bmatrix}, \quad \mathbf{\ddot{u}}_{12} = \begin{bmatrix} \mathbf{M}_{12} + \mathbf{M}_{12}^T & 0 \\ 0 & 0 \end{bmatrix}, \quad \mathbf{\ddot{u}}_{22} = \begin{bmatrix} \mathbf{M}_{22} & 0 \\ 0 & 0 \end{bmatrix} \]
(47a)

\[ \mathbf{\ddot{u}}_1 = \begin{bmatrix} \mathbf{C} \mathcal{E}^1 - \mathbf{C}_1 \\ -\mathbf{C}_2 \mathbf{M} \end{bmatrix}, \quad \mathbf{\ddot{u}}_2 = \begin{bmatrix} \mathbf{C} \mathcal{E}^2 - \mathbf{C}_2 \\ -\mathbf{C}_1 \mathbf{M} \end{bmatrix} \]
(47b)

\[ \mathbf{E} = \begin{bmatrix} \mathbf{C} \mathcal{E}^1 & \mathbf{C} \mathcal{E}^2 \\ \mathbf{C} \mathcal{G}^T & \mathbf{C} \mathcal{G}^T \end{bmatrix}. \]
(47c)

Matrices \( \mathbf{\ddot{u}}_{11}, \mathbf{\ddot{u}}_{12}, \mathbf{\ddot{u}}_{22} \) and \( \mathbf{E} \) are symmetric, whereas matrices \( \mathbf{\ddot{u}}_1 \) and \( \mathbf{\ddot{u}}_2 \) are skew-symmetric. By construction, the first \( 3N \) equations of system (45) are the local equilibrium equations of the problem and the last eight define the stress resultant as per equation (43). The first \( 3N \) equations also imply global equilibrium. Indeed, multiplication of system (45) by \( [\mathbf{Z}^T, \mathbf{0}] \) and the use of identities (37) yield the global equilibrium equations (41). Consequently, system (45) is solvable only for the stress resultants that satisfy the global equilibrium equations.

Mathematically, this solvability condition manifests itself by the fact that matrix \( \mathbf{E} \) is six times singular and its null space is
\[ \mathbf{N} = \begin{bmatrix} \mathbf{\ddot{u}} \end{bmatrix}. \]
(48)
Indeed, identities (38) imply $\hat{E}, N = 0$. The same identities also imply the following results:

$$\mathcal{A}^T \hat{H}_\alpha = \frac{1}{\xi} \left[ (C_G G)^T, \mathcal{G}_{R M} \right]$$

and $\mathcal{A}^T \hat{M} \omega \beta = \frac{1}{\xi} \left[ (\hat{\omega} M G)^T, 0 \right]$. 

5 DIMENSIONAL REDUCTION

The previous section has derived the governing equations for plates from three-dimensional elasticity directly. Rather than using a numerical approach, power series solutions of the problem will be obtained in a recursive manner.

5.1 Power series expansion

The stress resultants will be expanded in power series as

$$\hat{\mathcal{F}}^* = \sum_{k=0}^\varnothing \sum_{m+n=k} \beta^{(k)}_{m,n} \mathcal{L}_{m,n} = \sum_{k=0}^\varnothing \beta^{(k)}_{m,n} \mathcal{L}_{m,n}$$

(49)

where $\varnothing$ indicates the order of the series expansion and $\beta^{(k)}_{m,n} = \alpha_1^m \alpha_2^n / (m! n!)$ are the monomials of the expansion. The second equality defines a simplified notational convention that implies the summation over all values of indices $m$ and $n$ such that $m + n = k$, starting from $m = k, n = 0$. For the zeroth order, eq. (49) reduces to $\hat{\mathcal{F}}^* = \mathcal{F}^{(0)}$, where array $\mathcal{F}^{(0)}$ stores the stress resultants at $\alpha_1 = \alpha_2 = 0$. For the first order, $\hat{\mathcal{F}}^* = \mathcal{F}^{(0)} + \alpha_1 \mathcal{F}^{(1)} + \alpha_2 \mathcal{F}^{(1)}$, where arrays $\mathcal{F}^{(1)}_0$ and $\mathcal{F}^{(1)}_1$ store stress resultant gradients along unit vectors $b_1$ and $b_2$, respectively, at the same location. A total number of $N_f = 4(\varnothing + 1)(\varnothing + 2)$ stress resultant coefficients appear in the expansion.

Array $\mathcal{F}$ of size $N_f$, collects all the coefficients of the expansion as $\mathcal{F}^T = \{ \mathcal{F}^{(0)}_0, \mathcal{F}^{(1)}_0, \mathcal{F}^{(1)}_1, \ldots \}$, up to order $\varnothing$. For uniformity with subsequent developments, eq. (49) is recast as

$$\hat{\mathcal{F}}^* = \sum_{k=0}^\varnothing \beta^{(k)}_{m,n} \mathcal{L}_{m,n}$$

(50)

where all entries of matrix $\mathcal{J}_{m,n}^{(k)}$ of size $8 \times N_\varnothing$, vanish, except for an identity matrix, $\mathcal{L}_{8 \times 8}$, such that $\mathcal{J}_{m,n}^{(k)} = \mathcal{L}_{m,n}^{(k)}$. The right-hand side of system (45) is now expanded as

$$\mathcal{Q} = \sum_{k=0}^\varnothing \beta^{(k)}_{m,n} \mathcal{E}_{m,n}^{(k)} + \sum_{k=0}^\varnothing \beta^{(k)}_{m,n} \mathcal{O}_{m,n}^{(k)}$$

(51)

where the first and second terms correspond to the series expansions of the stress resultants ($\mathcal{E}_{m,n}^{(k)} = [0^T, \mathcal{L}_{m,n}^{(k)}]$) and externally applied loading ($\mathcal{O}_{m,n}^{(k)} = [0^{(k)}_0, 0^{(k)}_1]$), respectively.

5.2 Global equilibrium equations

The combined governing eqs. (45) are solvable for stress resultants that are in equilibrium only, but expansion (50) does not satisfy this requirement. Indeed, introducing this expansion into the global equilibrium eqs. (41) yields a set of algebraic equations,

$$\Pi \mathcal{F} = -\mathcal{A}$$

(52)

where matrix $\Pi$ and array $\mathcal{A}$ are defined as follows

$$\Pi = \begin{bmatrix} -\mathcal{E}^{*T} & \frac{1}{\xi} \mathcal{L} \mathcal{E}^{*T} & \frac{1}{\xi} \mathcal{L} \mathcal{E}^{*T} & \frac{1}{\xi} \mathcal{L} \mathcal{E}^{*T} \\ 0 & -\mathcal{E}^{*T} & 0 & 0 \\ 0 & 0 & -\mathcal{E}^{*T} & 0 \end{bmatrix}, \quad \text{and} \quad \mathcal{A} = \begin{bmatrix} \mathcal{E}^{(0)}_0 \\ \mathcal{E}^{(1)}_0 \\ \mathcal{E}^{(1)}_1 \end{bmatrix}.$$
Array $\bar{A}$ stores the coefficients of the series expansion of the resultant of the externally applied loads, $\bar{F}$. For conciseness, the above expressions are written for $\Theta = 1$. Matrix $\Pi$ and array $\bar{A}$ are of size $N_r \times N_f$, where $N_r = 3(\Theta + 1)(\Theta + 2)$, and $N_r \times 1$, respectively.

Because matrix $\Pi$ is of rank $N_r = 5\Theta(\Theta + 1) + 2 + (\Theta + 2)$, system (52) imposes conditions on arrays $\bar{F}$ and $\bar{A}$. First, the solvability condition of system (52) is $\bar{A}^T \bar{A} = 0$, where $\bar{A}$, of size $N_r \times (N_r - N_c)$, is the null space of $\Pi^T$. This imposes $(N_c - N_r)$ conditions on the $N_r$ entries of array $\bar{A}$, leaving $N_c$ independent coefficients, stored in array $\theta$, and $\bar{A} = \Omega \theta$. Next, in the absence of externally applied loads, $\Pi \bar{F} = 0$, and $\bar{F}$ must span the null space of $\Pi$, which is of size $N_r \times (N_r - N_c)$. Hence, $\bar{F}$ can be expressed in terms of $N_f = N_r - N_c$ independent coefficients, stored in array $\theta$, and $\bar{F} = \Delta \theta$. The complete solution of system (52) is now $\bar{F} = \Delta \theta + \bar{F}_0$.

For later developments, it is convenient to select the eight first entries of array $\theta$ as the stress resultants, $\bar{F}_{0,0}^{(0)}$, the remaining entries correspond to stress resultant derivatives. Matrix $\Delta$ then presents the following structure

$$\Delta_{N_f \times N_f} = \begin{bmatrix} \mathcal{I} & 0 \\ \mathcal{I} & \mathcal{I} \end{bmatrix}^{T}. \quad (54)$$

Although not uniquely defined, Boolean matrices $\Omega_{N_r \times N_c}$, $\mathcal{Y}_{N_r \times N_r}$, and $\mathcal{Y}_{N_f \times N_f}$ can be obtained from symbolic manipulation programs easily for specific choices of the independent parameters.

It the stress resultant coefficients, $\bar{F}$, are obtained numerically, for instance from a finite element code solving plate problems, system (52) will not be satisfied exactly. The stress resultant coefficients can then be corrected, in a least squares sense, to enforce the exact satisfaction of this system. The highest-order equilibrium equations suffer from a truncation error: for the $\Theta = 1$ case illustrated in eq. (53), $\mathcal{X}^{1+1} \bar{F}_{1,0} = \Delta_{1,1}^{(1)}$ and $\mathcal{X}^{1+1} \bar{F}_{1,0} = \Delta_{1,0}^{(1)}$. For the unloaded case, these imply the vanishing of the first derivatives of the shear force components, which instead, should be related to the second derivatives of the stress resultants, if not for the truncated expansion. To remedy this shortcoming, corrective loading resultants are evaluated as $\bar{A}_{1,0}^{(1)} = \bar{F}^T \bar{Q}_{1,0}^{(1)} = \mathcal{X}^{1+1} \bar{F}_{1,0}^{(1)}$ and applied to the plate as distributed loads, $\bar{Q}_{1,0}^{(1)} = \bar{Z} \bar{Q}_{1,0}^{(1)} = \mathcal{X}^{1+1} \bar{F}_{1,0}^{(1)}$. Of course, this correction is applied to the highest order only.

### 5.3 The series solution

The unloaded plate problem, i.e., $\bar{Q} = 0$, is considered first. Particular solutions of system (45) are obtained by assuming the solution array to be expanded in power series, $\bar{\varphi} = \sum_{\Theta=0}^\infty \beta_m^{(k)} \bar{X}^{(k)}_{m,n} \bar{F} = \sum_{\Theta=0}^\infty \beta_m^{(k)} \bar{X}^{(k)}_{m,n} \bar{F}$, where $\bar{X}^{(k)T} = [\bar{X}^{(k)}_{m,n} \bar{X}^{(k)}_{m,n+1} \ldots]$, and matching the coefficients of the monomials. Because array $\theta$ is arbitrary, the following recursive equations are obtained

$$\begin{align*}
\tilde{E} \bar{X}^{(\Theta)}_{m,n} &= \bar{F}^{(\Theta)} \\
\tilde{E} \bar{X}^{(\Theta-1)}_{m,n} &= \bar{F}^{(\Theta-1)} + \tilde{E} \bar{X}^{(\Theta)}_{m,n+1,n} + \tilde{E} \bar{X}^{(\Theta)}_{m,n-1,n+1}, \\
\tilde{E} \bar{X}^{(k)}_{m,n} &= \sum_{l=0}^k \bar{X}^{(k+l)}_{m,n+l} + \tilde{E} \sum_{l=0}^k \bar{Q}^{(k+l)}_{m,n+l} + \tilde{E} \sum_{l=0}^k \bar{Q}^{(k+l)}_{m,n+l+1} + \tilde{E} \sum_{l=0}^k \bar{Q}^{(k+l)}_{m,n+l+2}.
\end{align*} \quad (55c)$$

For simplicity, matrix $\tilde{E}$ that multiplies each term of these equations was omitted. Note the recursive nature of the equations where the solutions of systems (55a) appear on the right-hand side of systems (55b), etc. Equations (55a) and (55b) represent $\Theta + 1$ and $\Theta$ independent linear systems, respectively, for the values of indices $m$ and $n$ such that $m + n = \Theta$ and $m + n = \Theta - 1$, respectively. Equations (55c) are valid for $k = 0, 1, \ldots, \Theta - 2$; each equations represents a total of $k + 1$ linear systems for $m + n = k$. Matrix $\tilde{E}$ is six times singular and its kernel is given by eq. (48). Because the stress resultants are in equilibrium, it is proved easily that the solvability conditions are satisfied for each of the systems (55). More details about the solution process are given in appendix B.
The nodal forces defined by eq. (24) can be expressed as \( \tilde{P} = \sum_{k=0}^{\ell} \tilde{P}^{(k)} \tilde{Y}^{(k)} \), where matrices \( \tilde{Y}^{(k)} \) are defined as
\[
\tilde{Y}^{(k)} = \bar{M} \tilde{J}^{(k)} + \bar{C} \tilde{W}^{(k)},
\]
and matrices \( \tilde{J}^{(k)} \) as
\[
\tilde{J}^{(\ell)} = \tilde{G} \tilde{S}^{(\ell)}, \\
\tilde{J}^{(k)} = \tilde{G} \tilde{S}^{(k)} + \hat{\alpha}_1 \tilde{W}^{(k+1)} + \hat{\alpha}_1 \tilde{W}^{(k+1)},
\]
where eq. (57b) holds for \( k = 0, 1, \ldots, \ell - 1 \). The first \( 3N \) equations of each of systems (55) can be recast in the following form
\[
\tilde{E} \tilde{W}^{(\ell)} = -\tilde{C} \tilde{J}^{(\ell)}, \\
\tilde{E} \tilde{W}^{(k)} = -\tilde{C} \tilde{J}^{(k)} + \hat{\alpha}_1 \tilde{W}^{(k+1)} + \hat{\alpha}_1 \tilde{W}^{(k+1)},
\]
where eq. (58b) holds for \( k = 0, 1, \ldots, \ell - 1 \). Factorizing \( \bar{G} \) in the last eight equations of each of systems (55) yields the following relationships
\[
\bar{G} \tilde{Y}^{(k)} = \tilde{E}^{(k)},
\]

### 5.4 Global compliance matrix

Once the series solution of the combined equations has been found through the recursive solution of eqs. (55), the associated local strain components can be recovered from eq. (16). The displacement derivatives are \( \hat{u} = \tilde{a}^T \tilde{Z}^{(k)} \hat{w}_1 + \tilde{a}^T \tilde{Z}^{(k)} \hat{w}_2 + \tilde{a}^T \tilde{Z}^{(k)} \hat{w}_1 + \tilde{a}^T \tilde{Z}^{(k)} \hat{w}_2 = \tilde{a} (\tilde{w}^x - \tilde{w}^{(k)} \tilde{w}^x) + \tilde{a} \hat{w}_1 + \tilde{a} \hat{w}_2 \) and the displacements are \( \hat{u} = \tilde{Z}^{(k)} \tilde{w}^x + \tilde{w} \). Evaluating the local strains at \( \bar{a}_1 = \bar{a}_2 = 0 \) leads to
\[
\tilde{Y}^{x}_0 = \frac{\bar{E}}{\bar{M} \bar{Z}^{(0)} + \bar{C} \tilde{W}^{(0)}} \tilde{X}^{(0)} + \frac{\bar{E} \tilde{W}^{(0)}}{\bar{M} \bar{Z}^{(0)} + \bar{C} \tilde{W}^{(0)}} \tilde{W}^{(0)}.
\]

The second term vanishes because of eq. (33), and hence,
\[
\tilde{Y}^{x}_0 = \frac{\bar{E}}{\bar{M} \bar{Z}^{(0)} + \bar{C} \tilde{W}^{(0)}} \tilde{X}^{(0)}.
\]

where matrix \( \tilde{Z}^{(0)} \) stores the nodal displacements derivatives, \( \tilde{w}^{(0)} \), and nodal warping, \( \tilde{W}^{(0)} \),
\[
\tilde{Z}^{(0)} = \begin{bmatrix} \tilde{P}^{(0)} & \tilde{W}^{(0)} \end{bmatrix}.
\]

The global compliance matrix is obtained by evaluating the strain energy stored in the three-dimensional structure. Introducing the local strain distribution (60) into the strain energy density, \( \tilde{a} \), and integrating through the thickness of the plate yields
\[
\tilde{a} = \frac{1}{2} \bar{a}^T \tilde{S}^{x} \bar{a}.
\]

where \( \tilde{S}^{x} \) is the positive-definite, symmetric, global compliance matrix of the plate,
\[
\tilde{S}^{x} = (\bar{Z}^{x} \bar{a})^{-1} \begin{bmatrix} \bar{M} & \bar{C}^T \bar{E} \end{bmatrix} \bar{Z}^{x} \bar{a}.
\]

The global compliance matrix is of size \( N_l \times N_l \), i.e., depends on the order of the series expansion. Within the discretization and truncation errors, the strain energy computed by eq. (62) equals that computed in the local model exactly because it is obtained through integration of the local strain.

456
distribution. Clearly, the strain energy depends on the stress resultants (the first eight entries of array \( \theta \)) but also on their spatial derivatives (the remaining entries of array \( \theta \)). For higher-order series expansions, the strain energy depends on increasingly higher-order derivatives of the stress resultants.

Because they are based on the rigid-normal assumption, the Reissner-Mindlin plate theories involve \( 8 \times 8 \) compliance matrices that defines the strain energy per unit area of the plate as

\[
\bar{a} = \frac{1}{2} \bar{\mathbf{F}}^{*T} \bar{\mathbf{F}} \bar{\mathbf{F}}^{*},
\]

where the reduced compliance matrix, \( \bar{\mathbf{F}} \), stores the first \( 8 \times 8 \) entries of matrix \( \mathbf{F}^{*} \). Although of reduced size, this compliance matrix takes into account warping deformations, in contrast with the Reissner-Mindlin compliance matrix \( \bar{\mathbf{F}} \), which does not.

Given the special structure of matrix \( \mathbf{F} \) expressed by eq. (54), it can be shown that the reduced compliance matrix, \( \bar{\mathbf{F}} \), becomes independent of the order of the expansion for \( \theta > 1 \). Because this matrix captures the plate’s strain energy density more accurately than that derived from Reissner-Mindlin theories, the preferred expression for this strain energy density is

\[
\bar{a} = \frac{1}{2} \bar{\mathbf{F}}^{*T} \bar{\mathbf{F}} \bar{\mathbf{F}}^{*},
\]

where the reduced compliance matrix, \( \bar{\mathbf{F}} \), stores the first \( 8 \times 8 \) entries of matrix \( \mathbf{F}^{*} \). Although of reduced size, this compliance matrix takes into account warping deformations, in contrast with the Reissner-Mindlin compliance matrix \( \bar{\mathbf{F}} \), which does not.

5.5 Invariance of the solution

The local strain field and compliance matrix defined by eqs. (60) and (63), respectively, have been derived from particular solutions of the recursive equations (55). Indeed, due to the singularity of matrix \( \bar{\mathbf{F}} \), general solutions of systems (55) are in the form of \( \mathbf{X}^{(k)} = \mathbf{X}^{(k-1)} - \mathbf{F} \mathbf{A}^{(k-1)} \), where matrix \( \mathbf{A}^{(k)} \) stores arbitrary coefficients and matrix \( \mathbf{F} \), defined by eq. (48), the null space of matrix \( \mathbf{F} \). It is shown now that the local strain field and compliance matrix derived in the previous section are independent of \( \mathbf{A}^{(k)} \), i.e., the solution is invariant and physically meaningful.

First, the recursive solution of eqs. (55) implies

\[
\mathbf{X}^{(k)} = \mathbf{X}^{(k-1)} - \mathbf{F} \mathbf{A}^{(k-1)},
\]

where eq. (65b) holds for \( k = 0, 1, \ldots, \theta - 2 \) and the following matrices were defined, \( \mathbf{X}^{T} = \{0, 1\} \) and \( \mathbf{X}^{T} = \{0, 1\} \). For \( k = 0 \), eq. (65b) implies

\[
\mathbf{X}^{(0)} = \mathbf{X}^{(0)} + \mathbf{X}^{(0)} \mathbf{A}^{(0)} + \mathbf{X}^{(0)} \mathbf{A}^{(0)} + \mathbf{X}^{(0)} \mathbf{A}^{(0)}.
\]

Introducing this result into eq. (57) and using identities (32) yields

\[
\mathbf{F}^{(k)} = \mathbf{X}^{(k)} + \mathbf{G} \mathbf{X}^{(k)},
\]

Identity (38a) and eqs. (56) then prove the invariance of matrices \( \mathbf{F}^{(k)} \) and \( \mathbf{X}^{(k)} \).
expected on physical grounds: matrices \( \tilde{Y}^{(k)}_{m,n} \) yield the nodal forces, which remain unaffected by the addition of rigid-body motions.

For a general solution, the strain distribution given by eq. (60) becomes \( \gamma^f = \left[ (\tilde{A}N)/\tilde{Z}_0 + (\tilde{B}N)/\tilde{Z}_0 \right] \theta . \) Identity (18) then yields \( \gamma^f = \gamma^0 \), i.e., the local strains are also invariant, as expected. Finally, the three-dimensional stress tensor is obtained from the constitutive laws (19) as \( \tilde{Z}_0 = \tilde{Z}_0^* \). Because the three-dimensional stress is invariant, so is the corresponding strain energy, and hence, the compliance matrix defined by eq. (63) is also invariant.

6 STRESS RECOVERY

The previous section has focused on the determination of the plate’s stiffness matrix. The attention now turns to the recovery of the local stress and strain fields, which depend on the details of the applied loading. Hence, the right-hand side of system (45) must now include all applied loads and their power series expansion given by eq. (51). The solution procedure is based on a recursive process similar to that described in the previous section; details are omitted. The local strain components, defined in (60), are \( \gamma = \left[ \tilde{A}N \right] \tilde{Z}_0 \). Once local strains are recovered, local stresses follow from the constitutive laws (19).

As discussed in sec. 4.5, system (45) is solvable only when the global equilibrium equations are satisfied. In practice, the stress resultants are obtained from the numerical solution of the plate equations, using a finite element approximation, for instance. In such case, a least squares process is used to correct the stress resultants and their spatial derivatives to ensure strict satisfaction of the global equilibrium equations.

7 NUMERICAL RESULTS

To validate the dimensional reduction and stress recovery analysis procedures proposed in this paper, a set of numerical examples will be presented.

7.1 Cylindrical bending problem

This example focuses on a laminated composite plate of width \( L = L/4 \) undergoing cylindrical bending, as depicted in fig. 3. The plate is of infinite length along unit vector \( i \) and is simply supported along the two opposite edges at \( \alpha = 0 \) and \( L \). The plate is subjected to distributed transverse pressures \( p_i \alpha \) and \( \eta \), all made of the same material with the following physical properties: longitudinal, transverse, and shear moduli are \( E_I/E_T = 25, G_{LT}/E_T = 0.5 \), and \( G_{TT}/E_T = 0.2 \), respectively; Poisson’s ratios are \( v_{LT} = v_{TN} = 0.25 \). The stiffness parameters are non-dimensionalized with respect to transverse modulus \( E_T \).

Two lay-up configurations are investigated with the following stacking sequences starting from the bottom ply, case (a): \( [15\alpha, -15\alpha] \) and case (b): \( [30\alpha, -30\alpha, 30\alpha] \); \( 0\alpha \) fibers are aligned with unit vector \( i \) and a positive ply angle indicates a right-hand rotation about unit vector \( i \).

In the proposed approach, a single four-node one-dimensional element was used to model each ply and the investigation focuses on the stress recovery process. Pagano [32] obtained an analytical solution of this problem and exact stress resultants and their derivatives were obtained by integrating the exact three-dimensional stress distribution through the plate’s thickness.

The distributions of the non-dimensional stress components, \( \tilde{\sigma}_{ij} = \sigma_{ij}/p_0 \), through the thickness of the plate were evaluated at mid-span, i.e., at \( x_1 = L/2 \), for lay-up configurations (a) and (b). The predictions of the present approach are compared with the exact solution of Pagano [32], the solution of classical lamination theory [9, 10], and that resulting from the Variational Asymptotic Plate And Shell Analysis [25, 26] (VAPAS). Both the present approach and VAPAS used second-
order series expansion ($\ell = 2$).

Figure 4 shows the distribution of in-plane stress component, $\sigma_{11}$, through the thickness of the plate for lay-up configuration (a). The solid and dashed lines present the exact [32] and classical lamination solutions, respectively; the VAPAS and present solutions are indicated with symbols $\circ$ and $\times$, respectively. The predictions of the proposed approach are in excellent agreement with those of the exact solution. Figures 5 and 6 present the results for stress components $\sigma_{22}$ and $\sigma_{12}$, respectively. The corresponding results for lay-up (b) are presented in figs. 7 to 9. Finally, the transverse shear stress component, $\sigma_{13}$ and $\sigma_{23}$, are shown in figs. 10 and 11, respectively. For both lay-up configurations, the predictions of the proposed approach are in excellent agreement with the analytical solutions. Small discrepancies are observed for the derivatives of stress components, as expected from the truncated Taylor series expansion inherent to the proposed approach.

Finally, the stress component gradients, $\partial \sigma_{13} / \partial \alpha_1$ and $\partial \sigma_{23} / \partial \alpha_1$, are shown in figs. 10 and 11, respectively. For both lay-up configurations, the predictions of the proposed approach are in excellent agreement with the analytical solutions. Small discrepancies are observed for the derivatives of stress components, as expected from the truncated Taylor series expansion inherent to the proposed approach.

Figure 3. Configuration of the cylindrical bending problem.

Figure 4. Distribution of stress component $\sigma_{11}$ through the thickness, case (a).

Figure 5. Distribution of stress component $\sigma_{22}$ through the thickness, case (a).
**Figure 6.** Distribution of stress component $\bar{\sigma}_{12}/p_0$ through the thickness, case (a).

**Figure 7.** Distribution of stress component $\bar{\sigma}_{11}$ through the thickness, case (b).

**Figure 8.** Distribution of stress component $\bar{\sigma}_{22}$ through the thickness, case (b).

**Figure 9.** Distribution of stress component $\bar{\sigma}_{12}$ through the thickness, case (b).
7.2 Spatial bending problem

The square laminated composite plate of size $L \times L$ and thickness $h = L/10$ depicted in fig. 12 undergoes spatial bending. The plate is simply supported along all four edges and is subjected to distributed transverse pressures $p_1(\alpha_1, \alpha_2) = p_0(\alpha_1, \alpha_2) = p_0/2\sin(\pi\alpha_1/L)\sin(\pi\alpha_2/L)$ over both top and bottom surfaces (For clarity, fig. 12 only shows the loading acting over the top surface). The plate consist of 4 plies, each of thickness $t_p = h/4$. All four plies are made of the same material with the following stiffness properties: longitudinal and shear moduli are $E_L/E_T = 25$, $G_{LT}/E_T = 0.5$, and $G_{TT}/E_T = 0.2$, respectively; Poisson’s ratios are $\nu_{LT} = \nu_{TN} = 0.25$.

![Figure 12. Configuration of the spatial bending problem.](image)

The lay-up configuration investigated here presents the following stacking sequence, $[0^\circ, 90^\circ, 0^\circ]$; $0^\circ$ fibers are aligned with unit vector $\overrightarrow{t_1}$ and a positive ply angle indicates a right-hand rotation about unit vector $\overrightarrow{t_3}$.

In the proposed approach, a single four-node one-dimensional element was used to model each ply. Because the lay-up is symmetric through the plate’s thickness, only three sub-matrices of the complete $8 \times 8$ reduced stiffness matrix, $\mathbf{S}_{1d}$, do not vanish: the $3 \times 3$ in-plane, $2 \times 2$ shearing, and $3 \times 3$ bending stiffness matrices, denoted $\mathbf{A}_{1d}$, $\mathbf{S}_{1d}$, and $\mathbf{D}_{1d}$, respectively. The same holds for the stiffness matrix obtained from the Reissner-Mindlin theory, denoted $\mathbf{S}_{RM}$. The in-plane stiffness matrices predicted by the two approaches are

$$
\begin{align*}
\mathbf{A}_{1d} &= \begin{bmatrix}
1.303 & 0.025 & 0.0 \\
0.025 & 1.303 & 0.0 \\
0.0 & 0.0 & 0.050
\end{bmatrix}, & \mathbf{A}_{RM} &= \begin{bmatrix}
1.312 & 0.0336 & 0.0 \\
0.0336 & 1.312 & 0.0 \\
0.0 & 0.0 & 0.050
\end{bmatrix}.
\end{align*}
$$

While the two matrices differ, it is clear that the warping induced deformations taken into account by the present approach have minimal effect on the in-plane stiffness matrix. Although the discrepancies are larger, the same observations apply to the bending stiffness matrix,

$$
\begin{align*}
\mathbf{D}_{1d} &= \begin{bmatrix}
334.2 & -20.9 & 0.0 \\
-20.9 & 1838 & 0.0 \\
0.0 & 0.0 & 41.7
\end{bmatrix}, & \mathbf{D}_{RM} &= \begin{bmatrix}
340.3 & -28.0 & 0.0 \\
-28.0 & 1846 & 0.0 \\
0.0 & 0.0 & 41.7
\end{bmatrix}.
\end{align*}
$$

461
Finally, the same warping deformations affect the shearing stiffness matrix drastically,

\[
\frac{S_{\text{d}}}{10^{-2}E_T L} = \begin{bmatrix}
1.903 & 0.0 \\
0.0 & 3.179
\end{bmatrix}, \quad \frac{S_{\text{RM}}}{10^{-2}E_T L} = \begin{bmatrix}
3.500 & 0.0 \\
0.0 & 3.500
\end{bmatrix}.
\]

(69)

As expected, warping induced deformations reduce the plate’s effective stiffness.

Because the configuration of the present problem is simple, Navier type solutions [1, 2] can be found easily with a single sine-wave term only. Such solutions were obtained based on two stiffness matrices: the reduced stiffness matrix of the proposed approach, \(\bar{C}_{\text{d}}\), and that of Reissner-Mindlin theory, \(\bar{C}_{\text{RM}}\). Table 1 lists the maximum values of the stress resultants and the mid-point vertical displacement for these two cases. Clearly, the predictions based on the proposed stiffness matrix are in closer agreement with the exact solution than those based on the classical, Reissner-Mindlin stiffness matrix.

<table>
<thead>
<tr>
<th></th>
<th>Exact</th>
<th>Navier, (\bar{C}_{\text{d}})</th>
<th>Navier, (\bar{C}_{\text{RM}})</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Values</td>
<td>Difference</td>
<td>Values</td>
</tr>
<tr>
<td>(Q_1^*/p_0)</td>
<td>0.2357</td>
<td></td>
<td>0.2321</td>
</tr>
<tr>
<td>(Q_2^*/p_0)</td>
<td>0.0826</td>
<td></td>
<td>0.0863</td>
</tr>
<tr>
<td>(M_1^*/(p_0L))</td>
<td>-0.0221</td>
<td></td>
<td>-0.0231</td>
</tr>
<tr>
<td>(M_2^*/(p_0L))</td>
<td>0.0708</td>
<td></td>
<td>0.0695</td>
</tr>
<tr>
<td>(M_{12}^*/(p_0L))</td>
<td>0.0042</td>
<td></td>
<td>0.0043</td>
</tr>
<tr>
<td>(E_T u_3^*/(p_0L))</td>
<td>7.3698</td>
<td></td>
<td>7.6378</td>
</tr>
</tbody>
</table>

Table 1. Comparing the exact and Navier solutions

In section 5.4, the global compliance matrix was reduced from size \(N_t \times N_t\) to 8 \(\times\) 8, arguing that the magnitudes of the stress resultants gradients must be far smaller than those of the stress resultants themselves. To verify this claim, the strain energy density at the plate’s mid-point was evaluated based on the exact solution of the problem derived by Fan and Ye [33], to find \(a = 1/2 \int_{\Omega} \mathbf{F}^T \mathbf{Q} \mathbf{F} \, d\alpha_3 = 2.0625 \, p_0^2 L / E_T \). Next, the same strain energy was evaluated based on the reduced stiffness matrix, \(\bar{C}_{\text{d}}\), and stress resultants of the associated Navier solution, leading to \(a = 2.0618 \, p_0^2 L / E_T \). Finally, if using the Reissner-Mindlin compliance matrix, \(\bar{C}_{\text{RM}}\), and stress resultants of the corresponding Navier solution, the same energy becomes, \(a = 1.9933 \, p_0^2 L / E_T \).

The strain energy predicted by the proposed approach is in close agreement with its exact counterpart: the 0.03\% discrepancy is due to discretization and truncation errors. With the Reissner-Mindlin stiffness matrix, the error becomes 3.4\%. Clearly, the reduced stiffness predicted by the proposed approach should be used instead of its Reissner-Mindlin counterpart as it captures the strain energy density more accurately.

Next, using the predictions obtained from Navier’s solution with the proposed reduced stiffness matrix, the local stress field was evaluated through the thickness of the plate. In the stress recovery process, the predictions of four expansion orders, constant, linear, quadratic, and cubic, were compared to assess the convergence of the proposed approach.

Figures 13, 14, and 15 depict the through-the-thickness distribution of non-dimensional direct stress components, \(\sigma_i = \sigma_i / p_0, i = 1, 2, 3\), at the center of the plate. In these figures, the predictions for \(\theta = 0, 1, 2, \) and 3 are indicated with symbols \(\circ, \nabla, \Delta, \) and \(\times\), respectively, while the exact solution by Fan and Ye [33] is indicated by solid lines.

For reasons of symmetry, the shear stress components all vanish at the center of the plate. The proposed approach was used to evaluate the shear stress gradients at that location, \(\partial \overline{\sigma}_{ij} / \partial \overline{a}_i, \partial \overline{\sigma}_{23}/\partial \overline{a}_2, \) and \(\partial^2 \overline{\sigma}_{12}/(\partial \overline{a}_1 \partial \overline{a}_2)\) and the predictions are shown in figs. 16, 17, and 18, respectively. For the lowest-order expansion, \(\theta = 0\), these stress gradients all vanish and the predictions for \(\theta = 1 \) or 2 are not accurate. For \(\theta = 3\) the predictions of the proposed approach are in close agreement with the exact solutions, demonstrating its good convergence characteristics. Although
the plate equations were solved using Navier’s solution, very accurate stress and stress gradient distributions are recovered through the plate’s thickness using $\vartheta = 3$.

8 CONCLUSIONS

In this paper, the governing equations for three-dimensional plate problems, combining the equilibrium conditions of both local and global models, were derived from the principle of virtual work. Power series solutions of these equations were found. This rigorous approach to dimensional reduction yields the constitutive laws for the plate and accurate stress distributions through its thickness. Numerical examples were presented that illustrate the proposed approach and demonstrate its accuracy.

The proposed dimensional reduction process shows that the plate’s strain energy density depends on stress resultants and their spatial derivatives, in contrast with classical plate theories, for which the same strain energy is a function of stress resultants only. For plate problems, spatial derivatives of the stress resultants should be far smaller than the stress resultants themselves. This observation enables the derivation of an $8 \times 8$ stiffness matrix for the plate. This stiffness matrix is of the same size as that used in classical plate theories, although its entries differ, because they reflect the plate’s three-dimensional deformations.
In the proposed approach, the assumptions associated with commonly used plate theories have been eliminated altogether. Yet, the predicted three-dimensional stress distributions through the plate’s thickness compare favorably with exact elasticity solutions. In contrast with higher-order and layer-wise plate theories, the present approach does not increase the number of unknowns used in plate theories. In fact, the proposed approach can be used in conjunction with existing plate models, such as those implemented in commercial finite element packages. It provides an improved, $8 \times 8$ stiffness matrix for the plate and furthermore, accurate, through-the-thickness stress distributions can be obtained through a simple post-processing operation.

The proposed approach can be generalized to plates undergoing large displacements and rotations, but small strains, leading to a geometrically exact formulation that can be used for dynamics applications.

**Acknowledgments**: The authors gratefully acknowledge the support for this work from the National Natural Science Foundation of China (Grant No. 11272203)

### A BOOLEAN MATRICES

Several Boolean matrices used throughout the formulation are defined here. First, matrices $\mathbf{\bar{a}}_1$ and $\mathbf{\bar{a}}_2$ are defined as

$$
\mathbf{\bar{a}}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{\bar{a}}_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}.
$$

Next, matrices $\mathbf{\bar{b}}_1$ and $\mathbf{\bar{b}}_2$ are defined as

$$
\mathbf{\bar{b}}_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1/2 & 0 \end{bmatrix}, \quad \mathbf{\bar{b}}_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1/2 & 0 & 0 \end{bmatrix}.
$$

The following combinations will be used as well

$$
\mathbf{\bar{a}} = \begin{bmatrix} \mathbf{\bar{a}}_1 \\ \mathbf{\bar{a}}_2 \end{bmatrix}, \quad \mathbf{\bar{b}} = \begin{bmatrix} \mathbf{\bar{b}}_1 \\ \mathbf{\bar{b}}_2 \end{bmatrix}.
$$
Matrices $\hat{a}_1 = \text{diag}(a_1)$, $\hat{a}_2 = \text{diag}(a_2)$ and $\hat{a} = \text{diag}(a)$ gather the corresponding Boolean matrices at each node. A similar notion is used for the following matrices $\hat{b}_1 = \text{diag}(b_1)$, $\hat{b}_2 = \text{diag}(b_2)$ and $\hat{b} = \text{diag}(b)$. Note that the following identity exists $\hat{b} = \hat{a} = I_{5 \times 5}$ and $\hat{b} \hat{a} = \bar{I}_{5n \times 5n}$.

Another set of Boolean matrices, $\bar{L}_1$ and $\bar{L}_2$, are defined as

$$\bar{L}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad \bar{L}_2 = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{73}$$

Next, matrices $q_1$ and $q_2$ are defined

$$q_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1/2 & 0 \end{bmatrix}, \quad q_2 = \begin{bmatrix} 1/2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1/2 \end{bmatrix}. \tag{74}$$

The following combinations will be used as well

$$t = \begin{bmatrix} \bar{L}_1 \\ \bar{L}_2 \end{bmatrix}, \quad q = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}. \tag{75}$$

Note that the following identity exists

$$q^t = \bar{L}_{8 \times 8}. \tag{76}$$

## B SOLUTION OF THE RECURSIVE SYSTEM

The solution of recursive system (55) can be organized in a rational manner to minimize computational cost. First, matrices $h^{(0)}, h^{(1)}, \ldots, h^{(r)}$ are found using the following recurrence

$$\tilde{E} h^{(0)} = \bar{E}, \tag{77a}$$

$$\tilde{E} h^{(1)} = \tilde{F}_1 h^{(0)} + \tilde{F}_2 \begin{bmatrix} 0 \\ h^{(0)} \end{bmatrix}, \tag{77b}$$

$$\tilde{E} h^{(k)} = \tilde{F}_1 \begin{bmatrix} h^{(k-1)} \\ 0 \end{bmatrix} + \tilde{F}_2 \begin{bmatrix} 0 \\ h^{(k-1)} \end{bmatrix} + \bar{M}_{11} \begin{bmatrix} h^{(k-2)} \\ 0 \end{bmatrix} + \bar{M}_{12} \begin{bmatrix} 0 \\ h^{(k-2)} \end{bmatrix} + \bar{M}_{22} \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{77c}$$

The following short-hand notation was introduced: matrix $h^{(k)} = [h^{(k)}_{0,0}, \ldots, h^{(k)}_{k,k}]$ of size $N \times 8(k+1)$. All zero matrices appearing in eqs. (77) are of size $N \times 8$. The lowest order solution of recursive system (55) is then

$$X^{(0)}_{0,0} = \begin{bmatrix} h^{(0)}(0) \\ h^{(1)}(0) \\ h^{(2)}(0) \end{bmatrix} \Lambda, \tag{78}$$

where the vertical lines separate the orders of the expansion. At the first order, the solution is

$$X^{(1)}_{0,0} = \begin{bmatrix} 0 \\ h^{(0)}(0) \\ 0 \end{bmatrix} \Lambda, \tag{79a}$$

$$X^{(1)}_{0,1} = \begin{bmatrix} 0 \\ h^{(0)}(0) \\ 0 \end{bmatrix} \Lambda. \tag{79b}$$
At the second order, the solution is

\[
\Delta_{2,0}^{(2)} = \begin{bmatrix}
0 & 0 & 0 & \cdots & \Delta^{(0)} & 0 & 0
\end{bmatrix} \Delta \quad \text{(80a)}
\]

\[
\Delta_{2,1}^{(2)} = \begin{bmatrix}
0 & 0 & 0 & \cdots & \Delta^{(0)} & 0 & 0
\end{bmatrix} \Delta \quad \text{(80b)}
\]

\[
\Delta_{2,2}^{(2)} = \begin{bmatrix}
0 & 0 & 0 & \cdots & \Delta^{(0)} & 0 & 0
\end{bmatrix} \Delta \quad \text{(80c)}
\]

Higher-order solutions follow a similar pattern.

REFERENCES


Topology Optimization of Bearing Domains in Flexible Multibody Systems

Ali Moghadasi, Alexander Held, Robert Seifried

Institute of Mechanics and Ocean Engineering
Hamburg University of Technology
Eißendorfer Straße 42, 21073 Hamburg, Germany
[ali.moghadasi, alexander.held, robert.seifried]@tuhh.de

ABSTRACT

Topology optimization can be used for optimizing members of flexible multibody systems to enhance different characteristics of these systems. Here, a topology optimization scheme for flexible multibody systems is presented in which a more accurate model of bearing domains is included in the optimization. This is especially of interest since a connection between flexible members in a multibody system using bearings is widely seen in many application. Moreover, the modeling accuracy of the bearing area is shown to be influential on the shape of the optimized structure. As an application example, a flexible slider-crank is optimized here for minimum value of compliance in its nonrigid part. The flexible body is incorporated in the multibody simulation using the floating frame of reference formulation and its elastic deformation is approximated using shape functions calculated in the model order reduction analysis. The bearing loads are incorporated in this framework by introducing a corrector load in the bearing model. The results of the optimization is compared with an equivalent system which is modeled using nonlinear finite elements.

Keywords: Topology Optimization, Flexible Multibody System, Bearing Model.

1 INTRODUCTION

A critical issue in the design and utilization of machines such as industrial robots and mechanism is the energy consumption. One way to improve the energy efficiency is by reducing the moving masses in a dynamic system. However, this often results in the loss of stiffness of components and unacceptable elastic deformations and vibrations. In high-speed or high-precision systems, these deformations might deteriorate the required performance and accuracy.

In order to reduce the mass of the members of multibody system without hindering its performance, different optimization techniques have been proposed and adopted. Shape optimization of flexible multibody systems has been investigated in [1, 2, 3]. In [4, 5], the topology optimization of flexible multibody systems is discussed which are modeled using nonlinear finite element methods. An alternative approach is investigated in [6, 7, 8], where the floating frame of reference formulation is used to incorporate the flexible bodies in the multibody system.

In the former formulation, the nonlinear finite element model of the flexible body undergoes large motions and deformations in the dynamic simulation. In the latter, the motion of the flexible body is described by a body-related frame which undergoes large nonlinear motions and rotations with respect to the inertial frame of reference. Additionally, the small linear deformation of the flexible body is described in this body-related frame using global shape functions. Limiting the number of shape functions to a small set which describes the body deformation sufficiently well, the computational effort of the dynamic simulation is reduced. This is especially favorable in an optimization process where the dynamic simulation needs to be performed in every iteration. However, due to the linear approximation of the elastic body deformation, the modeling of contact as it occurs in bearings is not trivial.

Depending on the bearing type between flexible bodies, different nonlinear phenomena could occur in the contact area. For example, in a clearance joint, effects such as friction, impact and
separation of contact surfaces are to be seen [9]. Consequently, the contact forces in the real joints are considerably higher than the ones predicted by ideal joints, see [10, 11]. These effects could be considered if the dynamic system is modeled using nonlinear finite element methods. However, using the floating frame of reference formulation, considering these effects is more challenging.

The general approach regarding the bearing domain in the topology optimization of flexible bodies in multibody systems is to exclude this domain from the optimization. This can be achieved by modeling the joints as a set of cantilever beam elements [12] or using rigid planar elements [7]. This provides a simple joint model that represents some of the characteristics of the actual joint, however, the aforementioned nonlinear phenomena are not captured. Consequently, the deformations and stresses near the joints are not represented correctly.

In this work, a brief review of the existing modeling approaches for the bearing domain is presented. In addition, a modified approach is introduced which enhances the accuracy of the optimization results in the bearing domain. Two types of joints, namely clearance and non-clearance joints are discussed and the optimization results based on the modified approach are compared to the previous models as well as to nonlinear finite element models.

This paper is organized as follows. In Section 2 the optimization problem formulation and its steps are presented. In this scope the topology optimization, multibody dynamic simulation and the model reduction techniques are briefly explained. In Section 3, different modeling approaches for the joints in the optimization of multibody systems are reviewed and a modified approach is presented. The optimization results based on the application example of a slider-crank mechanism are shown and discussed in Section 4. At the end, the presented results are concluded in Section 5.

2 PROBLEM FORMULATION

In the following, the basic ideas of topology optimization of flexible multibody systems modeled with the floating frame of reference formulation, shown in Figure 1, are reviewed. At first, the floating frame of reference formulation and the modal reduction are briefly described. Then the iterative optimization procedure is explained.

2.1 Floating Frame of Reference Formulation

Elastic bodies can be incorporated in multibody systems using the floating frame of reference approach [18]. Here, the deformation of an elastic body is described with respect to a reference frame and the rigid body motions are described by the large non-linear motion of this reference frame. If the deformations are small and elastic, as they occur in typical machinery tools and robotics applications, the flexible bodies can be efficiently described using this approach. The position vector of a point \( P \) on the flexible body is determined by the sum of rigid motion and the linear deformation \( u_p \). The deformation \( u_p \) is approximated by a time-independent matrix of shape functions \( \Phi \) and time-dependent elastic coordinates \( q_e \) as

\[
u_p(R_P, t) = \Phi(R_P)q_e(t)
\]

where \( R_P \) is the position vector of the point \( P \) in the body-related frame.

The equations of motion in minimal coordinates are obtained by selection of generalized coordinates \( q = [q_r^T, q_e^T]^T \in \mathbb{R}^f \), whereby, \( q_r \in \mathbb{R}^f_r \) represents \( f_r \) rigid body degrees of freedom and the elastic coordinates \( q_e \in \mathbb{R}^f_e \) are the generalized coordinates describing the elastic deformations, see [19]. The equations of motion in minimal coordinates are consequently given as

\[
\begin{bmatrix}
M_{rr}(q) & M_{re}(q) \\
M_{er}(q) & M_{ee}(q)
\end{bmatrix}
\begin{bmatrix}
\ddot{q}_r \\
\ddot{q}_e
\end{bmatrix} +
\begin{bmatrix}
k_r(q, \dot{q}) \\
k_e(q, \dot{q})
\end{bmatrix}
\begin{bmatrix}
\dot{q}_r \\
\dot{q}_e
\end{bmatrix} +
\begin{bmatrix}
0 \\
K_{ee}q_e + D_{ee}\dot{q}_e
\end{bmatrix} =
\begin{bmatrix}
g_r(q, \dot{q}) \\
g_e(q, \dot{q})
\end{bmatrix} +
\begin{bmatrix}
B_r(q) \\
B_e(q)
\end{bmatrix}b.
\]

Here, the index \( r \) and \( e \) denote the association with rigid and elastic generalized coordinates respectively. In addition, \( M \in \mathbb{R}^{f_r \times f_r} \) is the generalized mass matrix, \( k \in \mathbb{R}^{f_r} \) is the vector of generalized
Coriolis, gyroscopic and centrifugal forces, $K_{ee}$ and $D_{ee} \in \mathbb{R}^{f_e \times f_e}$ are the stiffness and damping matrices of the elastic coordinates, $g \in \mathbb{R}^f$ is the vector of generalized applied forces, $B \in \mathbb{R}^{f \times p}$ is the input matrix and $b \in \mathbb{R}^p$ is the vector of control inputs. Rearranging the equation of motion (2) with respect to stiffness of elastic coordinates gives

$$K_{ee} \dot{q}_e = g_e + B_e b - D_{ee} q_e - k_e - M_{ee} \ddot{q}_e - M_{ee} \ddot{q}_e = f_{eqv} \quad (3)$$

whereby the right-hand side terms are denoted by the equivalent load $f_{eqv}$.

### 2.2 Model Reduction

The global shape functions $\Phi$, which are used in Equation (1) to approximate the body’s deformation, are often obtained from finely resolved finite element models. For the linear equations of motion of the finite element model it holds

$$M_e \ddot{u} + D_e \dot{u} + K_e u = f_e \quad (4)$$

with $M_e$, $D_e$, $K_e$ and $f_e$ being the mass matrix, damping matrix, stiffness matrix and the applied force vector, respectively. The inclusion of all the degrees of freedom of $u$ in the multibody simulation is computationally very expensive. Using the model reduction techniques, the number of degrees of freedom is reduced which significantly enhances the time efficiency of the dynamic simulation.

The shape functions $\Phi$ can be determined by performing a modal reduction. In this work, the model reduction technique based on the component mode synthesis is used [20]. With this model reduction technique, the shape functions are obtained from selected eigenmodes of the system as well as constraint modes, which are the static solutions of the flexible body when a unit displacement is applied to the fixed degrees of freedoms.

### 2.3 Topology Optimization

Topology optimization of mechanical elements and structures under static loads is well established, see [13]. For dynamic systems, the application of topology optimization is often limited to the optimization of single elastic bodies subjected to dynamic loads, see [14]. The optimization techniques which have been developed for static problems can be adapted and used to optimize members of a flexible multibody system which have large nonlinear motions.
For these systems, similar to the case of static structures, the material model is based on the power law model, i.e. the solid isotropic material with penalization (SIMP) approach. In this approach, the density and stiffness of intermediate elements are traditionally penalized by

\[ \rho_i = x_i \rho_0, \quad E_i = x_i^p E_0, \quad i = 1, \ldots, n \]

where \( p \) is the penalization factor, \( n \) is the number of elements, \( \rho_0 \) is the density and \( E_0 \) is the Young’s modulus of the solid material. Moreover, \( x \) is a density-like parameter which takes the value of 0 when an element is empty, and is equal to 1 for full elements. Using this approach, for the elements with small values of \( x \), the mass-to-stiffness penalization ratio goes to infinity which leads to spurious modes in the modal analysis, see [15]. Therefore, different modified models based on the power law in Equation (5) have been introduced [15, 16]. Based on the investigation of these models in the optimization of multibody systems [17], the alternative penalization method by Olhoff and Du in [16] is utilized. There, the mass penalization in changed for elements with small values of \( x \) as

\[ \rho_i = \begin{cases} x_i \rho_0 & 0.1 \leq x_i \leq 1 \\ s x_i^q \rho_0 & x_{\text{min}} \leq x_i < 0.1 \end{cases} \]

\[ E_i = x_i^p E_0, \quad i = 1, \ldots, n \]

where \( x_{\text{min}} \) is the minimum allowable value of \( x \). The constants \( s \) and \( q \) need to be accordingly assigned. The optimization is formulated as the minimization of the compliance \( c \) which results in a more stiff structure. The compliance is computed at several time points that have maximal deformation. The corresponding sensitivities are calculated with the assumption of design independent forces

\[ c(x) = u^T K_e u, \quad \frac{\partial c(x)}{\partial x_i} = -p(x_i)^{p-1} u_i^T K_{e0} u_i, \quad i = 1, \ldots, n. \]

Thereby, \( u \) is the global displacement vector, \( K_{e0} \) is the stiffness matrix of the \( i \)-th element without SIMP penalization and \( u_i \) is the displacement vector of the \( i \)-th element. The loads applied on the flexible structure in a multibody system are dependent on the design parameters but the assumption of design independent loads greatly reduces the computational cost of the sensitivity analysis. This simplification has to be evaluated carefully, for instance, if the loading of the flexible body results dominantly from its own inertia, this assumption is not anymore valid [17].

Following the objective function definition, the topology optimization can be formulated as

\[ \min_{x \in \mathbb{R}^n} : \quad c(x) = \sum_{j=1}^{m} u_j^T K_e u_j \]

subject to:

\[ h(x) = -V_0 + \sum_{i=1}^{n} x_i V_i \leq 0 \]

\[ 0 < x_{\text{min}} \leq x_i \leq 1, \quad i = 1, \ldots, n. \]

In this formulation, \( m \) is the number of time points at which the compliance is calculated and \( h(x) \) is the total mass constraint with \( V_0 \) being the total volume and \( V_i \) the volume of the \( i \)-th element. This constraint places a limit on the amount of material allowed to be distributed in the design domain. At any time point \( t = t_j \), the displacement vector \( u_j \) is computed using Equation (1) and the elastic coordinates \( q_e \), which have been determined in the time simulation of the multibody system.

### 3 Bearing Models

In the simulation of multibody systems, it is often assumed that the bearings between connecting bodies are ideal, meaning that it is frictionless and also the connecting points of two bodies are
coincident. However, these two points are not coincident when there is clearance in the bearing that connects the two bodies. The modeling of a bearing in the dynamic simulation is therefore dependent on the bearing type and the existence of clearance and friction in the joint. Here, three different modeling approaches for revolute joints in flexible multibody systems are examined. In Figure 2, an illustration of a revolute joint with exaggerated clearance is shown. The distance between the connecting nodes $O_1$ and $O_2$ is denoted by $\delta$ as a measure of clearance.

### 3.1 Ideal revolute joint

In an ideal revolute joint there are no deformation and no friction. This type of bearing can be modeled using a rigid domain in the connecting points of the flexible members. Examples for the optimization of flexible multibody systems using ideal bearings are found in [21, 22]. Introduction of rigid domains in the interface of connecting bodies in a flexible multibody system imposes the necessary condition for an ideal joint, since no deformation is allowed in the bearing domain. However, rigid domains are excluded from the topology optimization of flexible bodies using the optimization procedure described in the previous chapter. Therefore, using this approach, no information can be achieved about the optimum design of the rigid domain. In optimization examples where the design of the bearing domain is fixed or is not critical, utilization of rigid bearing domains could be considered.

### 3.2 Revolute joint with linear spring elements

One step to improve the modeling of the joint is to use linear spring elements [23]. For a revolute joint, the linear model can be implemented by a set of truss elements which connect the center interface node to the joint surface as shown in Figure 3. This linear model fits well in the finite element model which is then used in the flexible multibody simulation. Since the finite element model remains linear, standard model reduction techniques can be used.

In the topology optimization of a flexible body, it is important that the model of the joints generates a realistic distribution of the loads on the bearing surface. The load distribution affects the optimized design in the bearing domain as it is shown in Section 4. The model of linear truss elements is capable of depicting the relative motion of the connecting nodes. While this can be seen as a simple clearance model, it fails to generate the correct distribution of bearing loads. For a demonstration and comparison of the load distribution, a clamped pin-bearing geometry is considered in Figure 4 which is loaded with a vertical force of 1N on the center of the pin. The bearing has equal width and height of 0.03m, the pin radius is 0.008m and both the pin and bearing are made from aluminum with $E_0 = 7 \cdot 10^{10}$ N/m$^2$, $\rho_0 = 2750$ kg/m$^3$ and $\nu = 0.3$. In Figure 4a, the joint is modeled with linear trusses and in Figure 4b, it is modeled with gap elements which have zero stiffness under tension but are similar to linear truss elements when they are under compression. The gap elements are therefore able to model the separation of contact in the bearing. Comparing
Figure 4: Stress distribution (Pa) in a constrained joint with a) linear truss elements b) gap elements c) preloaded truss elements

Figure 4a and Figure 4b, apart from the obvious difference in stress distribution, the maximum stress in the Figure 4b is also considerably higher. The linear model does not only generate fictional stresses on the upper part of the bearing where separation occurs, it also underestimates the maximum stress in the structure. However, the gap elements lead to an overall nonlinear finite element model which could not be included in flexible multibody simulation using the floating frame of reference formulation and standard model reduction techniques.

In this scope, a modified approach is suggested in which the linear truss elements are used but the joint-structure contact is artificially preloaded. In the initial model, some of the truss elements are under tension, whereas in the preloaded model the truss elements are solely under compression. In the new loading condition, the pin is always in contact with the structure. Therefore this linear model can be used with introducing smaller errors to the simulation and optimization. As a comparison, in Figure 4c, the pin bearing geometry is modeled with the addition of equally distributed radial pressure. It can be observed that the preloaded joint generates a similar stress distribution in the bearing domain as the model with gap elements. However, as it would be shown in the next section, this model is still not accurate regarding the stress distribution in the bearing domain which is critical in the topology optimization of this domain.

3.3 Revolute joint with Hertzian contact

Alternative to the previous models, the nonlinearity of the contact in a joint considering the separation of the surfaces can be modeled using Hertzian contact law [24]. For two-dimensional contact of cylindrical bodies [25], the half contact width $b$ in Figure 2 is given as

$$b = \sqrt{\frac{4F}{\pi \left( \frac{1}{E_1} + \frac{1}{E_2} \right)}} \left( \frac{1}{R_1} - \frac{1}{R_2} \right)$$

(9)

where $\nu$, $E$ and $R$ are the Poisson’s ratio, Young’s modulus and the radius of the cylinder respectively. Additionally, $F$ is the load that is applied on the contact area. The pressure distribution in the contact area is parabolic and defined as

$$p(x) = p_{\text{max}} \sqrt{1 - \left( \frac{x}{b} \right)^2}$$

(10)

with the maximum pressure $p_{\text{max}}$ given as

$$p_{\text{max}} = \frac{2P}{\pi b}$$

(11)
The Hertzian contact law is a pure elastic model and does not consider the damping or the plasticity in the contact area. For extended models based on the Hertz law see [26]. Here, the classic formulation in Equations (9-11) is utilized. Using this formulation, it is possible to estimate the bearing loads more accurately, however, it is not possible to directly implement this in the simulation of the flexible multibody system with the floating frame of reference formulation since the approximation of deformation vector in Equation (1) is assumes linear behavior.

In order to enhance the accuracy of the model of the bearing domain, while still take advantage of the computationally efficient floating frame of reference formulation, a new corrector load $f_{\text{cor}}$ is calculated and added to the equivalent load $f_{\text{eqv}}$ in each time step. This additional load vector is defined as the difference between the bearing loads calculated with the linear model $f_{\text{lin}}$ and the one calculated with the contact model $f_{\text{con}}$ at each time step

$$f_{\text{cor}} = f_{\text{con}} - f_{\text{lin}}, \quad j = 1, 2, ..., m.$$  

(12)

Whereby, the rows of stiffness matrix $\bar{K}_e$ are equal to $K_e$ at the degrees of freedom of the nodes on the bearing surface and are 0 elsewhere. The load vector $f_{\text{lin}}$ contains the load distribution on the bearing surface when the model of joint with linear truss elements is used. The load vector $f_{\text{con}}$ contains the load distribution on the same nodes as for $f_{\text{lin}}$ but the distribution is calculated using the Hertzian contact law. For this calculation, the joint force $F$ is equal to the joint force when the model of linear truss elements is used. Including the correction loads of Equation (12) in the calculation of $u_j$ using Equation (1) gives the new displacement vectors

$$\dot{u}_j = \Phi q_{k,j} + K_e^{-1}f_{\text{cor}}, \quad j = 1, 2, ..., m.$$  

(13)

Thereafter, the calculation of objective function and gradients is straightforward using Equation (7) with $u$ replaced by $\dot{u}$. To demonstrate, the modified approach is used to model again the pin-bearing geometry shown in Figure 4 and the results are given in Figure 5. It can be clearly observed that the addition of correction loads will generate a more accurate stress distribution in the bearing domain considering that in Figure 5b, the same truss element model as in Figure 4a has been used.

4 OPTIMIZATION RESULTS

To test the modified joint modeling approach and compare it with other models, a topology optimization of a slider-crank mechanism is performed. The optimization results are validated with a nonlinear finite element model where the contact between the pin and the bearing domain is implemented directly in the dynamic simulation.
4.1 Application example

The application example of a slider-crank mechanism, shown in Figure 6, is used to demonstrate the effect of different bearing models on the topology optimization of flexible multibody systems. The slider-crank consist of a rigid crank and a flexible connecting rod which is connected to the sliding mass and the crank with revolute joints and is discretized with 9200 planar bilinear elements. The length of the connecting rod is 0.4 m and it is made from aluminum. The rigid crank is the driver arm and the crank angle $\alpha$ is given as $\alpha(t) = 8\pi t$. Moreover, the mass of the slider is set to 0.2 kg. The topology optimization is utilized to optimize the flexible connecting rod and the goal is to minimize the deviation from corresponding rigid system.

4.2 Optimized structures

The first case being examined is the rigid bearing model. Two different approaches are used for the rigid modeling of the bearing domain in the optimization of the flexible connecting rod. On the one hand, the bearing domain is completely replaced with a rigid domain and is therefore excluded from the optimization for which the result is shown in Figure 7a. On the other hand, a rigid ring is defined in the place of the revolute joint which results in the optimized structure shown in Figure 7b. Herewith, the bearing domain is included in the optimization, yet, the interaction between the pin and the bearing domain is neglected. This results in a joint which is barely included in the body. Moreover, based on these two results, it should be noted that a change in the model of the joint can strongly affect the overall optimized design of that structure.

In the next step, the revolute joints are modeled with truss elements. The contact surface of the pin and the bearing domain is also preloaded with an equally distributed radial pressure. The
Figure 8: Comparison of sliding mass deviation (left) and the compliance (right) using floating frame of reference approach (FFoR) and nonlinear finite element (nonlin)

Figure 9: Optimized design of bearing domain using floating frame of reference approach (FFoR) and nonlinear finite element (nonlin)

 optimized structure with this joint model is shown in Figure 7c, wherein, the joint is slightly more integrated in the structure compared to the model with rigid ring.

Next, the suggested addition of correction loads based on the Hertzian contact law is used. This strongly affects the design of the optimized connecting rod especially in the bearing domain. Here, a clearance of $\delta = 0.005R$ is assumed in the revolute joint with $R$ being the radius of the bearing. The optimized structure is shown in Figure 7d, in which, more material is shifted to the bearing domain and additional substructures are formed in this part. Moreover, the design of the left part of the connecting rod has slightly changed.

In order to compare the effect of correction loads in the final design of the bearing domain, the same connecting rod has been modeled and optimized using nonlinear finite element method. For this purpose, the simulation model in Figure 1 is implemented using the commercial code ABAQUS. The other optimization steps have been kept unchanged. The equivalence of the ABAQUS simulation model and the one based on the floating frame of reference formulation is first verified by comparing the sliding mass deviation in these two approaches using the same flexible slider-crank mechanism. The sliding mass deviation is the difference of the sliding mass position comparing the flexible and an equivalent rigid slider-crank mechanism. For this verification, the connecting rod is discretized excluding the bearing domain, and the slider deviation is computed, see Figure 8. Also, the compliance of the connecting rod is shown in Figure 8, which verifies the similarity of the deformations and dynamic behavior in the two simulation approaches.

By including revolute joints in the ABAQUS model, similar to the optimization using correction loads, a clearance of $\delta = 0.005R$ is assumed. The contact between the pin and the bearing domain is defined using the penalty method and the contact area is assumed to be frictionless. In Figure 9, the bearing domain design is depicted next to the bearing domain of Figure 7d. A qualitative comparison of these two designs shows similar formation of material when the linear model with addition of correction loads is used. For these results, it is important to consider the simulation times for the two different models. Using an equal number of elements and a similar problem setup.
in both models, the optimization using the floating frame of reference formulation takes roughly 3 hours, whereas the optimization based on ABAQUUS model takes about 110 hours to complete. The time efficiency of the former method which is due to the use of model reduction in the simulation of multibody system is greatly advantageous and the suggested modified joint model makes it more suitable for applications where the design of bearing domain is critical.

5 CONCLUSIONS

Modeling of joints and the design of bearing domains in the topology optimization of flexible multibody systems are discussed in this work. More specifically, the bearing domains are analyzed in flexible multibody systems modeled with floating frame of reference formulation. In this approach, since the model of flexible bodies is linear, implementation of joints and joint contact is challenging. Two modeling approaches of the joints in the optimization of dynamical systems are discussed which are the rigid and the linear interfaces. These models are easy to implement in the dynamic simulation and optimization, yet, they fail to represent the exact load distribution on the bearing domain, which leads to unrealistic design of the joints. In this regard, a modified approach based on the classic Hertzian contact law is presented.

This modified approach is implemented in an application example of a slider-crank mechanism. The optimized structure and in particular the bearing domains are in good agreement with the optimization results based on a nonlinear finite element model. The modified approach is shown to be more advantageous regarding its simulation time which is greatly lower than the latter approach.

REFERENCES


Towards viscoplastic constitutive models for Cosserat rods

Vanessa Dörlich#, Joachim Linn‡, Tobias Scheffer*, Stefan Diebels*

# Department Mathematical Methods in Dynamics and Durability
Fraunhofer Institute for Industrial Mathematics ITWM
Fraunhofer Platz 1, D-67663 Kaiserslautern, Germany
[vanessa.doerlich, joachim.linn]@itwm.fraunhofer.de

* Lehrstuhl für Technische Mechanik
Universität des Saarlandes
Campus A4 2, D-66123 Saarbrücken, Germany
[s.diebels, t.scheffer]@mx.uni-saarland.de

ABSTRACT
Flexible, slender structures like cables, hoses or wires can be described by the geometrically exact Cosserat rod theory. Due to their complex multilayer structure, consisting of various materials, viscoplastic behavior has to be expected for cables under load. Classical experiments like uniaxial tension, torsion or three-point bending already show that the behavior of e.g. electric cables is viscoplastic. A suitable constitutive law for the observed load case is crucial for a realistic simulation of the deformation of a component. Consequently, this contribution aims at a viscoplastic constitutive law formulated in the terms of sectional quantities of Cosserat rods. Since the loading of cables in applications is in most cases not represented by these mostly uniaxial classical experiments, but rather multiaxial, new experiments for cables have to be designed. They have to illustrate viscoplastic effects, enable access to (viscoplastic) material parameters and account for coupling effects between different deformation modes. This work focuses on the design of such experiments.

Keywords: Cosserat rods, viscoplasticity, multiaxial experiments, flexible structures, cables.

1 INTRODUCTION
Cables and hoses are slender flexible objects and can be described physically correctly by the geometrically exact theory of Cosserat rods [1]. The principal constituents of the rod model are geometrically exact kinematics relating configuration variables and objective strain measures, balance equations that govern the dynamic equilibrium of the sectional quantities, and constitutive equations, which yield the sectional forces and moments in terms of the deformation. Finding an appropriate constitutive model is especially necessary to enable a realistic simulation of the deformation behavior of a structure. A viscoelastic model formulated in the sectional force and moment quantities and objective deformation measures of the Cosserat rod model was already presented in [2] and [3]. In this contribution, we focus on viscoplastic effects and present first steps towards an application oriented modeling approach on the level of sectional quantities similar to Simo et al. [4].

Cables and hoses are components with a complex multilayer structure, which consists for example of parallel or twisted wires, insulating layers, woven fabrics and sheaths. Their behavior under load can thus be investigated on different scales, for example on the microscale if single wires are considered or on the macroscale by observing the deformation of whole cables. Due to various effects like friction between the constituents, pull-out of wires or delamination, a treatment on the microscale and subsequent coupling of the effects is computationally too complex. Furthermore, these effects cannot be measured in experiments on the microscale. Therefore, the deformation of cables and hoses will be investigated and modeled phenomenologically on the level of sectional quantities.
Since cables include a variety of materials like ductile metals, (hyper-)elastic polymers or brittle glass fibers, it is not sufficient to use linear-elastic constitutive models if a realistic simulation is the aim. Inelastic behavior like viscosity, plasticity and friction cannot be neglected. This can already be seen when classical experiments like uniaxial tension, torsion or three-point bending tests are executed cyclically, which will be shown in section 2.5. These classical tests provide information about the tensile, torsional and bending stiffness of the specimens. However, they do not represent realistic situations of cables in applications, which are mostly multiaxial and combine several of the classical load cases. Therefore, multiaxial experiments providing information about the coupling of the single stiffnesses have to be executed. Section 2.5.2 deals with the design of suitable multiaxial experiments for the characterization of cables and hoses. In a first approach, we test straight clamped cables and hoses in large deformation experiments combining bending, torsion and tension after van der Heijden et al. [5, 6]. Since these experiments will prove to be not useful for characterizing cables and hoses, new simplified multiaxial experiments for clamped cables are designed.

2 BASIC CONSIDERATIONS ON MODELING AND EXPERIMENTS

2.1 Kinematics of Cosserat rods

The kinematics of a Cosserat rod are described by its configuration variables, see figure 1. The centerline curve \( r(s) \) and the moving frame \( \hat{R}(s) \) are both a function of the arc length \( s \) in the reference configuration [7]

\[
s \mapsto r(s) \in \mathbb{R}^3, \quad (1)
\]

\[
s \mapsto \hat{R}(s) \in SO(3). \quad (2)
\]

The orthonormal set of vectors \( a^{(\alpha)} \) defines the cross section where \( a^{(1)} \) and \( a^{(2)} \) span the cross section and \( a^{(3)} \) is the normal vector. The material strain measures related to the configuration variables are

\[
K^{(\alpha)} := \langle a^{(\alpha)}, a^{(3)} \times \partial_s a^{(3)} \rangle; \quad \Gamma^{(\alpha)} := \langle a^{(\alpha)}, \partial_s r \rangle; \quad \alpha = 1, 2
\]

\[
K^{(3)} := a^{(2)} \cdot \partial_s a^{(1)}; \quad \Gamma^{(3)} := \langle a^{(3)}, \partial_s r \rangle - 1. \quad (3)
\]

\( K^{(\alpha)} \) measures the bending curvatures for \( \alpha = 1, 2 \) and \( K^{(3)} \) the torsional twist. The transverse shear strain components are given by \( \Gamma^{(\alpha)} \) for \( \alpha = 1, 2 \) and the longitudinal strain is given by \( \Gamma^{(3)} \).
An extensible Kirchhoff rod satisfies the additional constraint

\[ t = \frac{\partial_s r}{\|\partial_s r\|} = a^{(3)} \]  

(4)

for the tangent vector \( t \) of the centerline, which inhibits transverse shearing by keeping the cross sections of the rod normal to the tangent vector [8]. Consequently, the components of \( \Gamma \) are

\[ \Gamma^{(\alpha)} \equiv 0; \ \alpha = 1, 2 \]  

(5)

\[ \Gamma^{(3)} = \|\partial_s r\| - 1. \]  

(6)

### 2.2 Static equilibrium equations

The system of static equilibrium equations for Cosserat rods

\[ \partial_s f + f_{\text{ext}} = 0 \]
\[ \partial_s m + \partial_s r \times f + m_{\text{ext}} = 0 \]  

(7)

is valid independent of the constitutive equations. It has to be satisfied by the spatial sectional forces \( f \) and moments \( m \), which can be resolved with respect to the moving frame according to

\[ f = \hat{R}(s) \cdot F; \ \ m = \hat{R}(s) \cdot M. \]  

(8)

### 2.3 Constitutive laws

The constitutive equations relate the material sectional forces \( F \) and moments \( M \) with the strain measures given in equations (3). A linear-elastic constitutive law formulated in the material sectional quantities is for example

\[ F = C_F \cdot \Gamma_{\text{el}}; \ \ M = C_M \cdot K_{\text{el}} \]  

(9)

with the effective stiffness matrices \( C_F \) and \( C_M \).

### 2.4 Formulation of plastic constitutive laws

The simplest case of a rate-independent elasto-plastic constitutive equation can be derived from figure 2 according to [9]. It consists of an elastic spring with Young’s modulus \( E \) and a Coulomb friction element with yield stress \( \sigma_y \). Due to the serial connection, the total strain \( \epsilon \) can be split into an elastic and a plastic part

\[ \epsilon = \epsilon^{el} + \epsilon^{pl}. \]  

(10)

The elastic stress on the spring is

\[ \sigma = E \epsilon^{el} \]  

(11)

and can be formulated with equation (10) as

\[ \sigma = E (\epsilon - \epsilon^{pl}). \]  

(12)
The plastic strain $\varepsilon^{pl}$ is an additional kinematic variable following the evolution equation

$$\dot{\varepsilon}^{pl} = \frac{\partial}{\partial t} \varepsilon^{pl} = \gamma \frac{\partial f}{\partial \sigma}$$

(13)

assuming that $\varepsilon^{pl}$ is a function of time $t$. The evolution equation is called an associative flow rule, if the potential relationship in equation (13) holds for the yield function $f$, $\gamma$ is a plastic multiplier. The yield function serves as a criterion for determining the occurrence of yield at a certain load. Algorithmic approaches to solve elasto- or viscoplastic problems can be found exemplarily in [9, 10].

Similar to equation (12), rate-independent elasto-plastic constitutive laws can be formulated in terms of the sectional forces and moments as

$$F = C_F \cdot (\Gamma - \Gamma^{pl}); \quad M = C_M \cdot (K - K^{pl}).$$

(14)

The long-term goal of this work is to extend this formulation to rate-dependent viscoplastic constitutive laws for Cosserats rods in terms of sectional quantities.

Remark: The primary interest lies in plastic bending and torsion of cables, since these cases occur more often in applications. Longitudinal extensional and transversal shear strains are assumed as approximately elastic, such that plastic effects herein can be neglected for practical reasons.

2.5 Experimental characterization of cables and hoses

Executing suitable experiments is an important part of modeling the deformation behavior of structural elements. Such experiments have to provide access to the model parameters and have to serve as an appropriate database for simulating a certain load case. The work of several authors has already shown for example that in order to perform multiaxial simulations, it is not sufficient to determine the model parameters in uniaxial experiments [11, 12, 13, 14]. In the first part of this section, classical experiments for the characterization of beam-like structures will be described, which enable access to linear-elastic stiffnesses. They already illustrate the necessity of including inelastic effects in the modeling of cables under load, when they are performed cyclically. However, these experiments are not sufficient to provide information about the coupling of different load cases and do not represent realistic loading of cables in application. Suitable multiaxial experiments combining different load cases have to be designed and executed in order to get information about the coupling of tension, torsion and bending. The second part of this section deals with this issue.

2.5.1 Classical experiments

Classical experiments for the characterization of beam-like structures available in the literature are for example uniaxial tension, torsion and three-point bending tests [15]. Figure 3 shows the schematic setups of the executed experiments. The experimental results can be analyzed under the assumption of linear-elastic behavior from the plot of the measured quantities $y$ and $x$, see figure 4. The tensile stiffness $(EA)_{uni}$, the torsional stiffness $(GJ)_T$ or the bending stiffness $(EI)_B$ can be calculated respectively by a linear relation according to

$$y = kx$$

(15)

from the linear slope $k$ as shown in table 1. The stiffnesses $(EA)_{uni}, (GJ)_T$ and $(EI)_B$ are rather component specific parameters than material parameters. They include information about the geometry of the cable in form of the cross section area $A$, the area moment of inertia $I$ and the polar moment of inertia $J$ in addition to the material parameters $E$, the Young’s modulus, and $G$, the shear modulus.

These classical experiments already enable access to the stiffnesses for linear-elastic constitutive models. By executing several consecutive cycles, they even provide information about the inelastic behavior of cables.
2.5.2 Multiaxial experiments

In order to get information about the coupling of different load cases, it is necessary to test cables in multiaxial experiments.

Van der Heijden et al. [5, 6] describe the writhing of cylindric metallic wires as multiaxial experiments for clamped rods. The experimental procedure is shown in figure 5. The ends of the specimen are clamped in one axis, the specimen is twisted about this axis and afterwards one clamp is moved towards the other clamp. These experiments are executed for several torsion angles. After Euler’s Theory of the elastica [16, 17], primary buckling of an untwisted specimen of length $L$ and bending stiffness $(EI)_B$ into a planar configuration occurs at a critical load $T$ according to

$$t = \frac{TL^2}{4\pi^2(EI)_B} = -1. \quad (16)$$

Pretwisted specimens will buckle directly into a spatial configuration where bending, torsion and - in certain cases - tension couple at another critical load. The ratio of bending to torsional stiffness determines the transition from planar to spatial configurations and vice versa. The final shape of the specimen at maximum displacement $D$ for $D/L = 1$ is a ring for any torsion angle. Stable as well as unstable configurations exist, since different buckling modes are possible.

The spatial configurations as well as the axial forces and moments are measured during the experiment. Van der Heijden et al. compare their experimental results for nitinol wires to semi-analytical computations using an inextensible Kirchhoff rod model (i.e. a constrained variant of the Cosserat rod with inhibited longitudinal extension and transverse shearing). They observe a characteristic shape of the experimental and simulated curves resulting from a normalization of the measured forces on the critical buckling load and the displacement $D$ on the original length, see figure 6. Besides, curves for different original lengths coincide for the same torsion angle.
3 CHARACTERIZATION OF A COAXIAL CABLE

3.1 Experiments

A coaxial cable is characterized using the described classical and multiaxial experiments. The cross section of the cable has a diameter of 2.8 mm and is shown in figure 7.

The cyclic loading in the classical experiments is applied displacement-controlled and 5 cycles are performed in each experiment. The maximum loading is 100 N for the uniaxial tension test, 360° for the torsion test and 6 mm for the three-point bending test. The results of the cyclic classical experiments on a coaxial cable are shown in figure 8. The stiffnesses calculated from the loading paths of the experiments are summarized in table 2, where the values for the first cycle are calculated separately from cycles 2 – 5. This illustrates that especially the torsional and bending stiffnesses change after the first cycle. All three experiments have in common, that the behavior is clearly non-linear and that hysteresis loops appear. The cyclic experiments also give proof of remaining plastic deformation. In the uniaxial tension test, a plastic stretch occurs, which results in a compressive force in order to reach the original length of the specimen. Similarly, a negative torsional moment is necessary during unloading in the torsion test because of a permanent twist. During three-point bending, the punch loses contact to the specimen while unloading at approximately 3 mm due to remaining curvature, see figure 8 d). In case of the uniaxial tension test, a sharp kink at approximately 54 N is visible in the first loading path, which can be interpreted.
Figure 5: Experimental procedure of multiaxial experiment: A straight specimen is clamped (left), the upper clamp is rotated by $4\pi$ (middle) and afterwards moved down by 60mm (right).

as a yield point for the tensile load case. To determine if this kink is the tensile yield point for this cable, cyclic uniaxial tension with a maximum axial force of 40 N has been executed. The result is shown in figure 8 in comparison to the loading up to 100 N. The expectation is being confirmed because in this case, no yield and nearly no hysteresis appears. Therefore, the yield point was not reached in this experiment.

Table 2: Stiffnesses measured in the classical experiments.

<table>
<thead>
<tr>
<th></th>
<th>$(EA)_{uni}$ [N]</th>
<th>$(GJ)_T$ [N·m²]</th>
<th>$(EI)_B$ [N·m²]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st cycle</td>
<td>2.020 · 10⁴</td>
<td>8.235 · 10⁻⁴</td>
<td>9.813 · 10⁻⁴</td>
</tr>
<tr>
<td>average of cycles 2 – 5</td>
<td>2.081 · 10⁴</td>
<td>1.078 · 10⁻³</td>
<td>1.898 · 10⁻⁴</td>
</tr>
</tbody>
</table>

The experimental setup by van der Heijden et al. described in section 2.5.2 is used to test the coaxial cable in a realistic multiaxial loading. Pretorsion angles of $0$, $\pi$, $2\pi$, $3\pi$ and $4\pi$ are applied before the upper clamp is moved by 60 mm. Specimens of two different lengths (100 mm and 180 mm) are used. The results are shown in figure 9. The left plot shows the results for a specimen of an initial length of 180 mm for different torsion angles. The shape of the curves differs already qualitatively from the theoretical results shown in figure 6. The right hand figure compares the results for both initial lengths for no pretorsion and $4\pi$ pretorsion. In both cases, the measured curves do not coincide after the normalization on length and critical buckling load. These results show that this type of experiment is too complex and its interpretation too complicated to be useful for determining the coupling between the different stiffnesses. Consequently, new simplified multiaxial experiments have to be designed which provide access to the desired information. The experimental procedures of the simplified multiaxial experiments are illustrated in figure 10. To determine the influences of pretorsion and prestretch, different pretorsion angles $\theta_T$ from $0^\circ$ to $360^\circ$ or different prestretches up to 100 N are applied.

The influence of pretorsion on the behavior of the coaxial cable under tension is shown in figure 11. The uniaxial tensile force $F_{uni}$ is plotted versus the displacement $D$ for different pretorsion angles. The tensile stiffness decreases with increasing pretorsion angles. This effect can be interpreted as a result of delamination and hence softening of the cable during the torsion. This result has to be
Figure 6: Theoretical results of multiaxial experiments for different torsion angles of a) 0, b) $\pi$, c) 2$\pi$, d) 3$\pi$, e) 4$\pi$, after [5] (left) and own results of simulation based on a discrete Cosserat rod model (right). Dashed lines symbolize instable branches, dots mark transition from stable to unstable branches and the circle marks the ring configuration.

Figure 7: Cross section of investigated coaxial cable.

handled carefully. In a zoomed in plot of the loading path of the first cycle for 0° and 360° it is visible that uniaxial tension force is already applied during the torsion step, see figure 11. Due to the twist, the specimen is elongated effectively, which results in an axial force. Consequently, the softening effect may be a result of the torsion induced pretension and is not a result of pure pretorsion. Therefore, the experimental procedure is slightly adapted in order to avoid this effect. A pure pretorsion load is achieved by activating force control during the pretorsion, which keeps the axial force at zero by adjusting the distance between the clamps. Besides, force control is already activated during the clamping of the specimen in order to avoid tensile forces on the specimen before testing. The adapted experimental procedure and resulting curves of uniaxial tension force are shown in figure 12. The same tendency is visible. An increasing pretorsion causes a decreasing tensile stiffness. Now, this result is only due to the combination of pure pretorsion and uniaxial tension. The pretorsion influences another inelastic feature, the hysteresis loops grow larger with increasing pretorsion, which is equivalent to an increasing dissipation. The aforementioned kink in the first loading path of the uniaxial tension is as well visible for the pretwisted specimens.

The influence of prestretch on the torsion behavior of the coaxial cable is shown in figure 13. The torsional moment $M_T$ is plotted versus the torsion angle $\theta_T$ for different prestretches up to a certain maximum force. No influence of different prestretches on the size of the hysteresis loops or the slope is visible. Consequently, the torsion behavior and plasticity of this cable are not a function of prestretch.
Figure 8: Results of cyclic classical experiments on a coaxial cable: a) uniaxial tension test, b) uniaxial tension including test for yield point, c) torsion test, d) three-point bending.

Figure 9: Results of multiaxial experiment on coaxial cable. Left: Comparison of different torsion angles for specimen of length 180 mm. Right: Comparison of results for specimens of both lengths for torsion angles of 0 and 4π.

487
Figure 10: Experimental procedure of multiaxial experiments combining tension and torsion. Left: A pretorsion of 180° is applied before cyclic uniaxial tension. Right: The specimen is prestretched until $F_{uni} = 100\text{N}$ before cyclic torsion is applied.

Figure 11: Left: Influence of pretorsion on the behavior in cyclic tensile test. Right: Loading paths of the first cycles for $0^\circ$ and $360^\circ$ pretorsion.

Figure 12: Left: Adapted experimental procedure in order to achieve pure torsion. Right: Influence of pure pretorsion on behavior in cyclic tensile test.
3.2 Simulation

Similar to van der Heijden et al. [5, 6], simulations of these experiments based on our implementation of a Cosserat rod have been performed for the material parameters of nitinol. The results are shown in figure 6 in comparison to the theoretical curves of van der Heijden. These results, obtained based on an elastic constitutive law, show good agreement to the theoretical results. Since the experimentally measured curves deviate not only quantitatively, but also qualitatively from this theoretical shape, it is necessary to include viscoplastic effects in the modeling of the multiaxial deformation of cables.

4 CONCLUSIONS & OUTLOOK

The results of the classical and multiaxial experiments show the necessity of including viscoplastic effects in the description of cables. Since these effects have to be measured and observed in experiments at first, suitable experimental procedures enabling access to the material parameters have to be developed. Three kinds of multiaxial experiments have been executed in this work. In the case of viscoplastic cables, the interpretation of the writhing experiment is too complicated and will thus be used as a verification experiment. However, the combination of cyclic tension and torsion is a first step towards the design of interpretable multiaxial experiments for the characterization of viscoplastic cables. Similar experiments combining torsion and bending will be designed in future work, because the primary interest lies in viscoplasticity herein. Furthermore, suitable constitutive laws in accordance to the observed hysteresis curves have to be designed. They will be adapted in analogy to three-dimensional constitutive laws but formulated in the terms of sectional kinetic quantities of Cosserat rods.
REFERENCES


490
A mixed shooting and harmonic balance method for mechanical systems with dry friction or other local nonlinearities

Frederic Schreyer, Remco I. Leine

Institute for Nonlinear Mechanics
University of Stuttgart
Pfaffenwaldring 9, 70569 Stuttgart, Germany
remco.leine@inm.uni-stuttgart.de, frederic.schreyer@inm.uni-stuttgart.de

ABSTRACT
In this paper we present a mixed shooting – harmonic balance method for large linear mechanical systems with local nonlinearities. The standard harmonic balance method (HBM), which approximates the periodic solution in frequency domain, is very popular as it is well suited for large systems with many states. However, it suffers from the fact that local nonlinearities cannot be evaluated directly in the frequency domain. The standard HBM performs an inverse Fourier transform, then calculates the nonlinear force in time domain and subsequently the Fourier coefficients of the nonlinear force. The disadvantage of the HBM is, that strong nonlinearities are poorly represented by a truncated Fourier series. In contrast, the shooting method operates in time-domain and relies on numerical time-simulation. Set-valued force laws such as dry friction or other strong nonlinearities can be dealt with if an appropriate numerical integrator is available. The shooting method, however, becomes infeasible if the system has many states. The proposed mixed shooting–HBM approach combines the best of both worlds.

Keywords: Shooting method, Harmonic Balance Method, local nonlinearities, periodic solutions.

1 INTRODUCTION
Finding periodic solutions of mechanical systems is an important task in the design process of machines and mechanical devices. For instance, knowledge of the response of the system to harmonic excitation is essential to obtain information about high cycle fatigue behaviour. In engineering systems local nonlinearities are present due to contact or coupling elements. These local nonlinearities can have a strong impact on the global system behaviour. Therefore, the nonlinearities have to be considered in the design process and must be modeled accurately as well as in a computationally efficient way.

The most popular methods to find periodic steady-state responses of nonlinear differential equations are the Harmonic Balance Method (HBM) [5] [6] and the Shooting Method [7]. The standard HBM approximates the periodic solution in frequency domain and is very popular as it is well suited for large systems with many states. Local nonlinearities cannot be evaluated directly in the frequency domain. The standard HBM performs an inverse Fourier transformation, and then calculates the nonlinear force in time domain and subsequently the Fourier coefficients of the nonlinear force. This procedure is often denoted as the Alternating Frequency Time Method (AFT) [4]. The disadvantage of the HBM is that strong nonlinearities are poorly represented by a truncated Fourier series. In contrast, the shooting method operates in time-domain and relies on numerical time-simulation. Set-valued force laws such as dry friction or other strong nonlinearities can be dealt with if an appropriate numerical integrator is available. The shooting method, however, becomes infeasible if the system has many states. The proposed mixed shooting–HBM approach combines the efficiency of HBM and the accuracy of the shooting method and has therefore many advantages.

In this paper the mixed shooting-HBM approach is introduced as a novel method to calculate periodic solutions of forced mechanical systems. Two different variants of the mixed shooting-HBM approach, which are called Method 1 and Method 2 in the following, are presented. Depending on
the position of the local nonlinearities within the mechanical system, the one or the other is better suitable. The more general Method 2 is tested on a multi-mass oscillator at the end of the paper and is compared to the full HBM and full shooting method. As local nonlinearities, dry friction as well as a hard unilateral constraint are investigated.

2 Mixed shooting-HBM approach

The mixed shooting-HBM approach uses the local character of the nonlinearities to find periodic solutions of mechanical systems efficiently. Therefore the system must be divided into linear and nonlinear subsystems. This can be done in two different ways which are defined in this paper as Method 1 and Method 2. First the system description is given and subsequently both methods are discussed.

2.1 System description

We consider a Lagrangian system of the form

\[
\dddot{\mathbf{q}}(t) + \dot{\mathbf{C}}\mathbf{q}(t) + \mathbf{K}\mathbf{q}(t) = \mathbf{f}_{\text{ex}}(t) + \mathbf{f}_{\text{nl}}(\mathbf{q}(t), \dot{\mathbf{q}}(t)),
\]

(1)

where \( \mathbf{f}_{\text{nl}} \) contains the nonlinear forces and \( \mathbf{f}_{\text{ex}}(t) = \mathbf{f}_{\text{ex}}(t+T) \) is the periodic forcing. We assume that the system consists of three subsystems with the generalized coordinates

\[
\mathbf{q} = \begin{pmatrix} q_1 \\ q_2 \\ q_3 \end{pmatrix},
\]

(2)

that the nonlinear forces only act on Subsystem 1, and that the system matrices \( \mathbf{M}, \mathbf{C}, \mathbf{K} \) have the following structure

\[
\mathbf{M} = \begin{pmatrix} M_{11} & M_{12} & M_{13} \\ M_{21} & M_{22} & M_{23} \\ M_{31} & M_{32} & M_{33} \end{pmatrix}, \quad \mathbf{f}_{\text{nl}}(\mathbf{q}, \dot{\mathbf{q}}) = \begin{pmatrix} f_{\text{nl}}(q_1, \dot{q}_1) \\ 0 \\ 0 \end{pmatrix}.
\]

(3)

Subsystem 1 is subjected to nonlinear forces, which only depend on its own positions and velocities, and is connected to Subsystem 3 through Subsystem 2, e.g. the three DOF oscillator shown in Figure 1.

2.2 Method 1

This first approach can only be applied to systems which satisfy the condition

\[
M_{31} = M_{13} = K_{31} = K_{13} = C_{31} = C_{13} = 0
\]

(4)

and is suitable for the following relation of the dimensions of the subsystems:

\[
\text{dim}(\mathbf{q}_3) \gg \text{dim}(\mathbf{q}_1) > \text{dim}(\mathbf{q}_2)
\]

(5)
For Subsystem 2 and 3 we use a harmonic balance approach and impose (as a numerical approximation) perfect constraints on the system which force the response to be harmonic of the form

\[
\begin{align*}
q_2(t) &= \hat{q}_2^0 + \sum_{k=1}^{n_H} \hat{q}_{2k}^k \cos k\omega t + \hat{q}_{2k}^k \sin k\omega t = V_+ (t)^T \hat{q}_2, \\
q_3(t) &= V_+ (t)^T \hat{q}_3,
\end{align*}
\]

with

\[
V_+ (t) = \begin{pmatrix}
I & \cos(\omega t)I & \sin(\omega t)I & \ldots & \cos(n_H \omega t)I & \sin(n_H \omega t)I
\end{pmatrix}.
\]

The Fourier coefficients of the generalized coordinates \(q_i(t)\) with \(i = 1, 2, 3\) are obtained from

\[
\hat{q}_i = \frac{2}{T} \int_0^T V_-(t) q_i(t) dt, \quad V_- (t) = \begin{pmatrix}
\frac{1}{2}I \\
\cos(\omega t)I \\
\sin(\omega t)I \\
\vdots \\
\cos(n_H \omega t)I \\
\sin(n_H \omega t)I
\end{pmatrix},
\]

with \(\omega = \frac{2\pi}{T}\) and \(n_H\) denoting the number of considered harmonics. The identity matrix \(I\) has here the dimension \(\dim(q_i)\). The motion \(q_1(t)\) of Subsystem 1 is described in time domain and is \textit{not} constrained to be harmonic. The equations of motion of Subsystem 2 and 3 can therefore be expressed in frequency domain as

\[
\begin{align*}
H_{21} \hat{q}_1 + H_{22} \hat{q}_2 + H_{23} \hat{q}_3 &= \hat{f}_{ex2}, \\
H_{32} \hat{q}_2 + H_{33} \hat{q}_3 &= \hat{f}_{ex3},
\end{align*}
\]

where \(H_{ij}\) are the dynamic stiffness matrices

\[
H_{ij} = \text{diag}(J_{ij,0}, J_{ij,1}, \ldots J_{ij,n_H})
\]

with

\[
J_{ij,k} = \begin{pmatrix}
-M_{ij}(k\omega)^2 + K_{ij} & C_{ij}k\omega \\
-C_{ij}k\omega & -M_{ij}(k\omega)^2 + K_{ij}
\end{pmatrix}.\]

Using (10) the Fourier coefficients \(\hat{q}_3\) can be expressed in \(\hat{q}_2\) as

\[
\hat{q}_3 = H_{33}^{-1}(\hat{f}_{ex3} - H_{32} \hat{q}_2)
\]

and can therefore be eliminated from the equations of motion in frequency domain, i.e.

\[
H_{21} \hat{q}_1 + (H_{22} - H_{23} H_{33}^{-1} H_{32}) \hat{q}_2 = \hat{f}_{ex2} - H_{23} H_{33}^{-1} \hat{f}_{ex3}.
\]

The equations of motion of Subsystem 1 are nonlinear and are simulated in time-domain. For known \(\hat{q}_2\) one can calculate its time-domain representation \(q_2(t)\) and its derivatives and solve the differential equation for \(q_1(t)\)

\[
\begin{align*}
M_{11} \ddot{q}_1(t) + C_{11} \dot{q}_1(t) + K_{11} q_1(t) &= -(M_{12} \ddot{q}_2(t) \\
+ C_{12} \dot{q}_2(t) + K_{12} q_2(t)) + f_{ex1}(t) + f_{n1}(q_1(t), \dot{q}_1(t))
\end{align*}
\]

using numerical integration techniques. In particular, if the nonlinear force \(f_{n1}\) is a dry friction force or, more generally, described by a set-valued force law, then dedicated time-integration schemes such as timestepping methods [1] [3] have to be used. Here it should be noted, that the
system (1) and consequently (15) turns into a differential inclusion if a set-valued force law is considered.

A periodic solution of the system can be represented by the trajectory \( q_1(t) \) on the interval \( 0 \leq t \leq T \) and by the Fourier coefficients \( \tilde{q}_2 \), as \( \tilde{q}_1 \) is expressed by (13). The initial condition \( q_1(0) \) and \( \dot{q}_1(0) \) together with \( q_2(t) = V_\perp(t)^T \tilde{q}_2 \) allow to construct \( q_1(t) \) over one period. The vector of unknowns

\[
x = \begin{pmatrix} \tilde{q}_2 \\ q_1(0) \\ \dot{q}_1(0) \end{pmatrix}
\]  

therefore fully represents a periodic solution of the system. Similar to a shooting method, we require for Subsystem 1 the periodicity conditions \( q_1(T) - q_1(0) = 0 \) and \( \dot{q}_1(T) - \dot{q}_1(0) = 0 \), where the state at \( t = T \) is obtained through numerical time-integration of (15). The periodicity conditions of Subsystems 2 and 3 are given in frequency domain by (14) and (13). Hence, we seek the zeros of

\[
f_R(x) = \begin{pmatrix} H_{22} \tilde{q}_1 + (H_{22} - H_{32} H_{33}^{-1} H_{32}) \tilde{q}_2 - \tilde{f}_{ex2} + H_{23} H_{33}^{-1} \tilde{f}_{ex3} \\ q_1(T) - q_1(0) \\ \dot{q}_1(T) - \dot{q}_1(0) \end{pmatrix}.
\]  

The zeros of \( f_R(x) \) can be solved with a Newton-type method by iterating

\[
x^{i+1} = x' - \left( \frac{\partial f_R}{\partial x} \right)^{-1} f_R(x').
\]  

### 2.3 Method 2

Alternatively, we can divide the system only into two parts, a linear and a nonlinear subsystem, where

\[
q_L = \begin{pmatrix} q_2 \\ q_3 \end{pmatrix}, \quad q_N = q_1.
\]  

The system matrices \( M, C, K \) and the nonlinear forces have then the following structure

\[
M = \begin{pmatrix} M_{NN} & M_{NL} \\ M_{LN} & M_{LL} \end{pmatrix}, \quad f_{nl}(q_N, \dot{q}_N) = \begin{pmatrix} f_{nlN} \\ 0 \end{pmatrix}.
\]  

This approach is more general than Method 1. Subsystem 1 and 3 do not have to be uncoupled since the system is not restricted to condition (4). The use of Method 2 can reduce the computational effort for systems for which the relationship \( \dim(q_L) \gg \dim(q_N) \) between the dimensions of the subsystems holds. Similar to Method 1, the motion of the linear subsystem is approximated by a truncated Fourier series

\[
q_L(t) = q_0^L + \sum_{k=1}^{nL} \tilde{q}_L^{0,k} \cos(k \omega t) + \tilde{q}_L^{1,k} \sin(k \omega t) = V_\perp(t)^T \tilde{q}_L.
\]  

Substituting this approximation into (1), the Fourier coefficients \( \tilde{q}_L \) of the linear subsystem can be expressed in the Fourier coefficients \( \tilde{q}_N \) of the nonlinear subsystem

\[
\tilde{q}_L = H_{LL}^{-1}(\tilde{f}_{exL} - H_{LN} \tilde{q}_N).
\]  

The equation of motion of the linear subsystem is therefore completely described by (22) and only the equation of motion of the nonlinear subsystem has to be described in the time domain. Using (22) together with (21), the time-evolution \( q_L(t) \) and its derivatives are given by \( \tilde{q}_N \). Hence, a differential equation with a reduced dimension

\[
M_{NN} \ddot{q}_N + C_{NN} \dot{q}_N + K_{NN} q_N = M_{NL} \ddot{q}_L + C_{NL} \dot{q}_L + K_{NL} q_L - f_{exN} + f_{fric}
\]  

\[494\]
has to be solved for \( q_N(t) \) using numerical time integration.

With (22) and (23) it is possible to represent a periodic solution of the full system in the unknowns

\[
x = \begin{pmatrix} \hat{q}_N \\ q_N(0) \\ \dot{q}_N(0) \end{pmatrix},
\]

where \( x \) is a zero of the residuum

\[
f_R(x) = \begin{pmatrix} \hat{q}_N - \text{FFT}(q_N(t)) \\ q_N(T) - q_N(0) \\ \dot{q}_N(T) - \dot{q}_N(0) \end{pmatrix}.
\]

Note that \( \text{FFT}(q_N(t)) \) is the Fourier transformation (9) of the solution of the differential equation (23) and \( \hat{q}_N \) are the Fourier coefficients which represent the dynamical behaviour of the linear subsystem through (22). If \( \hat{q}_N - \text{FFT}(q_N(t)) = 0 \) holds, then the linear subsystem is oscillating in correspondence to the movement of the nonlinear subsystem.

The iteration scheme of the mixed shooting-HBM approach (Method 2) with a Newton-type method is depicted in Figure 2. Note that, if \( \dim(q_L) = 0 \), then the method reduces to the standard shooting approach.

### 3 Numerical examples

The three DOF-oscillator (Figure 1) is used as a numerical benchmark to compare the mixed shooting-HBM approach (Method 2) with the full shooting method and the full HBM, in both computation effort as well as accuracy. Since the full and the mixed shooting-HBM approach solve the nonlinear subsystem as a nonlinear differential inclusion, modern time-stepping methods with a set-valued force law are used for both methods. In contrast to the full and mixed shooting-HBM, the standard HBM with alternating frequency time approach only calculates the nonlinear force in time domain which makes it impossible to use the same contact model. Two types of contacts are considered separately in this work to compare the different methods for a system which is subject to friction or to a completely elastic unilateral constraint.

#### 3.1 System with friction

First, the different methods are investigated for a system under influence of dry friction. Using the mixed shooting-HBM or the shooting approach a set-valued force law can be used within the concept of (measure) differential inclusions. The friction force is expressed by the set-valued relationship

\[
-\lambda_T \in \begin{cases} 
\mu F_N, & \gamma_T > 0, \\
[-1,1] & \gamma_T = 0, \\
-\mu F_N, & \gamma_T < 0.
\end{cases}
\]

The parameters \( \mu \) and \( F_N \) are the friction coefficient and normal load, respectively. This friction model cannot be used for the HBM because the problem is not solved in time domain. To compare the methods in a most suitable way, the friction force for the HBM is approximated using an arctangent function

\[
-\lambda_T^{\text{smooth}} = \mu F_N \frac{2}{\pi} \arctan(\kappa \gamma_T),
\]

being a smoothed approximation of (26). The approximation (27) tends to the set-valued force law (26) for large values of the smoothing parameter \( \kappa \), see Figure 3.
In Figure 4 the displacements of the system calculated with all three methods for the period $T = 10s$ are shown. During this period the first mass shows a pronounced stick-slip behaviour. Though for the Harmonic Balance Method 20 harmonics and for the mixed shooting-HBM only 3 harmonics are considered, the mixed method approximates much better the results of the full shooting method. The smoothing parameter is chosen preferably high ($\kappa = 800$). The mixed and full shooting method employ the set-valued description (26) of the friction law and can therefore describe stiction precisely. The HBM, however, not only uses the smoothed friction law (27) but also uses harmonic shape functions to approximate the friction force which leads to a poor description of this force. In contrast, the mixed shooting-HBM describes the whole nonlinear subsystem in time domain and approximates only the coupling between both subsystems with harmonic shape functions.

The mixed shooting-HBM approach becomes more advantageous than the full shooting method if the dimension of the linear subsystem is much larger than that of the nonlinear subsystem. To demonstrate this, the linear subsystem is extended with additional masses. This expanded model is used to compare the full HBM, the full shooting and the mixed approach. The excitation force is chosen as $f_{exi} = 0$ for $i = 1 \ldots n - 1$ and $f_{exn} = 5\cos(\omega t)$. The methods are compared for one excitation frequency in computation effort and accuracy. To start the calculation for a specific excitation frequency, a starting guess for the first iteration is needed. However, the methods iterate in
Figure 3: Set-valued friction force and approximated friction force for different values of the smoothing parameter $\kappa$.

Figure 4: Displacement and friction force for a periodic solution with period time $T = 10s$ of the three DOF oscillator with dry friction.
different unknowns and the same starting guess can therefore not be given. To provide comparable starting guesses, solutions for an excitation frequency close to the actual frequency are used as starting vectors for the iterative loops of the respective approximation methods.

In Figure 5 the relative error of the amplitude of the first and \(n\)th mass and the calculation effort is shown for different numbers of considered harmonics \(n_H\). Both ratios are with respect to the full shooting method, which is chosen as reference as it is almost exact. The results show that the computation effort for a moderate accuracy can be reduced drastically by using the mixed shooting-HBM approach. Compared to the HBM, the mixed approach shows for all values of \(n_H\) more accurate results. The horizontal plateau of the relative error of the mixed method can be explained by the limited resolution of the used Fourier transformation and the integration schemes. Therefore, the increasing number of considered harmonics reduces the error only to a specific value.

The used parameters for the calculations in this chapter are summarized in Table 1.

Table 1: Selected parameters for the system with friction.

<table>
<thead>
<tr>
<th>parameter</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>(m_i)</td>
<td>1</td>
</tr>
<tr>
<td>(k_i)</td>
<td>1</td>
</tr>
<tr>
<td>(c_i)</td>
<td>0</td>
</tr>
<tr>
<td>(\mu)</td>
<td>0.8</td>
</tr>
<tr>
<td>(\omega)</td>
<td>(\pi/2)</td>
</tr>
<tr>
<td>(f_{ex,30})</td>
<td>5(\cos(\omega t))</td>
</tr>
</tbody>
</table>

### 3.2 System with unilateral constraint

In the second example, the friction force in the first mass is replaced by a unilateral constraint. The unilateral constraint is modeled within the concept of measure differential inclusions using the hard contact law

\[
0 \leq g_N \perp \lambda_N \geq 0,
\]

where \(g_N\) is the gap \((g_N = g_{N,0} - q_1)\) and \(\lambda_N\) represents the contact force. The Newtonian impact law is expressed through the inequality complementarity

\[
0 \leq \gamma_N + e_N \gamma_N \perp \Lambda_N \geq 0 \quad \text{with} \ 0 \leq e_N \leq 1,
\]
with the post-and pre-impact relative velocities $\gamma^+_N$ and $\gamma^-_N$, the contact impulse $\Lambda_N$ and the restitution coefficient $e_N$. For a more detailed description of the contact law and impact law see e.g. [3]. The concept of measure differential inclusions with set-valued contact and impact law can only be used for the mixed shooting-HBM and full shooting method. As discussed in Section 3.1, the HBM only allows a smoothed contact law. Therefore, the contact for the HBM is modeled using a one-sided spring-damper element

$$-\lambda^\text{smoothed}_N = \begin{cases} k_cg_N + d_c\gamma_N & g_N \leq 0 \\ 0 & g_N > 0. \end{cases}$$

(30)

The equivalent restitution coefficient $e_N$ for a specific one-sided spring-damper element can be calculated following Brogliato [2]. Since, only a non-dissipative, elastic contact ($e_N = 1$) is used in this work $d_c$ is defined as zero and the model tends to the hard contact if $\lim k_c \to \infty$.

In Figure 6, the displacements of a five DOF oscillator with a gap ($g_N, 0 = 0.1$) at the first mass for the HBM, mixed shooting HBM and the full shooting method are depicted. Figure 7 shows the velocity of the first mass. The used parameters are summarized in Table 2. Like before for the system with dry friction, the HBM has difficulties to approximate the jump in the velocity of the first mass at the collision time-instant ($t = 0.55$), although the contact stiffness $k_c$ is chosen relatively high. The mixed and full shooting method show a true velocity jump whereas the HBM only gives a rough approximation of this phenomenon.

Table 2: Selected parameters for the system with impact.

<table>
<thead>
<tr>
<th>parameter</th>
<th>$m_1$</th>
<th>$m_{2-5}$</th>
<th>$k_i$</th>
<th>$c_i$</th>
<th>$k_c$</th>
<th>$e_N$</th>
<th>$\omega$</th>
<th>$f_{ex,5}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>value</td>
<td>10</td>
<td>1</td>
<td>1.3</td>
<td>0.3</td>
<td>8000</td>
<td>1</td>
<td>$\frac{\pi}{2}$</td>
<td>$5\cos(\omega t)$</td>
</tr>
</tbody>
</table>
Figure 6: Displacements of a five DOF oscillator with impact calculated with the different methods.

4 Concluding Remarks

The presented mixed shooting-HBM approach shows good characteristics in accuracy as well as in calculation effort, at least for the investigated benchmark system. Depending on the system size and the nonlinear characteristics the method can be a good alternative to the commonly used methods like HBM and shooting. It should be noted, that the numerical efficiency of the methods are hard to compare and that there exist alternative HBM methods to compute periodic solutions of systems with dry friction and impact. Further research will focus on providing a better comparison of the mixed shooting-HBM method with the existing methods.
Figure 7: Velocities of the 5 DOF oscillator with impact.

References


A proposition of analysis method for the system with contact between largely deformable body and rigid body

Taku Chida*, Yoshiki Sugawara*

*Faculty of Mechanical Engineering
Akita University
1-1 Tegata-Gakuen-machi, Akita-shi, Japan
ysugawara@mech.akita-u.ac.jp

ABSTRACT

In this paper, we propose a method for analyzing the dynamics between rigid body and largely deformable body which have contact with each other. The proposed method uses two analysis methods, one is material point method and the other is method for analysis of simultaneous multiple contact problem proposed by Pfeiffer et al. Behavior of the deformable body is derived by the use of material point method, which composes the deformable body as particle assembly. When the deformable body has contact with the rigid body, contact force has to be derived and determination of state transition associated with contact is also required. Such a contact force and state transition are obtained by solving linear complementarity problem which corresponds to the method proposed by Pfeiffer. Then, once contact force and state transition are derived, these information can be utilized in the analysis of deformable body by material point method. Therefore, influences of contact on the behavior of deformable body considered in the analysis and additionally the influence on the rigid body are also considered. The proposed method utilizes the material point as representative contact point, hence search of contact point is not required and consequently calculation cost becomes small. In this paper, the proposed method is introduced and numerical analysis is performed in order to study the feature of the proposed method.

Keywords: Large deformation, Unilateral Contact, Material Point Method, linear Complementarity Problem

1 INTRODUCTION

In general, wheel drive systems are adopted for locomotion of planetary exploration probes due to their mature technologies. On the other hand, jumping system is also gathering a lot of attractions because driving efficiency of the wheel drive system becomes worse on microgravity astronomical object. In such an environment of microgravity, it is possible to achieve high jump with low energy, however, there is a problem that any optimal jumping method is not established. Therefore, interaction between the probe and ground is quite important for achievement of optimal jump, that is to say, contact between the probe and the ground has to be analyzed.

There are some contact models for rigid bodies, and "Kelvin-Voigt model" [1] and idea of unilateral contact proposed by Pfeiffer et al. (here in after called "Pfeiffer model") [2] are generally well known. Kelvin-Voigt model consists of spring and dashpot and advantage of the model is its simplicity. However, there are some disadvantages and typical one is incorrectness of contact force at the moment of detachment, and the incorrectness results from the dashpot. Pfeiffer model deals with the relationships between the contact force and relative acceleration of bodies, and between impulse and relative velocity as sets of linear complementarity problem (LCP), which leads to effective computation. Furthermore, one of the typical features of Pfeiffer model is accurate representation of the state transitions in the case of simultaneous multiple contact.

Finite Element Method (FEM) is commonly-used method for investigation of the behavior of deformable bodies. However, it is quite difficult to determine the contact point when contact
problems between deformable bodies are dealt with. On the other hand, analysis method based on particle is gathering a lot of attentions in recent years. In the particle-based method, bodies to be represented are discretized. Hence, when those methods are applied to the contact problem between deformable bodies, it is expected that determination of contact point is not so difficult because those methods can assume particles to be the representative points for contact. There are also several types of particle-based method for analysis of largely deformable body, and, “Distinct element method : DEM [3]”, “Smoothed Particle Hydrodynamics : SPH [4]” and “Material Point Method : MPM [5]” are generally well known. DEM represent the deformable bodies by set of particles and model of their interaction between particles is given by spring and damper. The model of DEM is quite simple, but large number of particles is required for reasonable analysis results. SPH is also kind of method which deals with deformable bodies as particles. Particle in SPH is given by function called “kernel”, and interaction between particles is considered by overlap of those functions. In recent years, SPH is often used as strong method for analysis of deformable body, however there are some remaining problems in SPH, for example, instability in analysis, longer time required for searching of contact particles and so on. MPM also represents the analyzed object as several sets of particle which is called material point. MPM differs from other particle-based methods in that the method has background grid. The grid is used for update of parameters associated with dynamics, and calculation of update is performed based on the conventional FEM method. This is one of the advantages which MPM has, because various FEM techniques are applicable to analysis by the use of the background grid. Furthermore, another advantage is that existence of particles enables the MPM to analyze large deformation problems, because grid can be reset in each time step by keeping the information of deformation in particles [5].

In this study, we propose an analysis method that combines the above-mentioned MPM and Pfeiffer model. In the proposed method, material points defined in MPM are assumed to be representative point of the continuum bodies which have contacts with rigid bodies, and contact problems are solved using LCP if the contact between the representative points and rigid bodies are detected. As a result, we can calculate the contact forces and impulses acting between rigid and flexible bodies in an effective manner. Then, the behaviors of the rigid body, i.e. the probe, and the deformable body, i.e. the ground, are analyzed by the use of the derived forces and impulses.

This paper is organized as follows. In Section 2, the proposed analysis method is shown with the brief explanation of original Pfeiffer model and original MPM. In section 3, fundamental numerical analysis is performed in order to discuss the validity and property of the proposed method. Finally, conclusions and future works are described in Section 4.

2 PROPOSEDMETHOD

2.1 Analyzed object
Figure 1 shows the fundamental and typical analyzed object in this study. The system consists of one rigid body and one deformable body. In the following section, system in Figure 1 is used to develop the proposed method.

2.2 Proposed method

In order to utilize MPM, a number of material points are assigned to deformable bodies and background grid is introduced in the area in which the deformable body may move. Dynamics of each material point is calculated based on USF (Update Stress First) which is one of MPM. On the other hand, dynamics of the rigid body is calculated based on simple Newton-Euler equation. As mentioned in previous chapter, material point is assumed as representative point for contact. Hence, contact between rigid body and material point are investigated in each time step and contact problem is solved by the use of Pfeiffer model if contacts occur between the rigid body and some of material points. Then, contact force and state derived by solving the contact problem is used to calculate the dynamics of the system in each time step. Details of the proposed method with mathematical expression are shown in the following. Note that the details of MPM and Pfeiffer model are omitted in order to make the idea of the proposed method clear-cut in this paper and the references [2] and [5] should be referred for details of them.

Suppose that rigid body and deformable body are not in contact with each other at initial state and contact occurs after some time step. Such a state transition from non-contact state to contact state corresponds to the collision between both bodies and such transition is formulated by the use of Pfeiffer model. In order to apply Pfeiffer model to the system in this study, collision is divided into two phases, i.e. “Compression phase” and “Extension phase” [2]. Then, LCP is derived for Compression phase as

\[ y_c = A_c x_c + b_c \]
\[ y_c \geq 0 ; x_c \geq 0 ; y_c^T x_c = 0, \]

where \( A_c \) is a matrix which consists of inertia properties and Jacobian matrix between relative and generalized velocity, \( b_c \) is a vector which includes relative velocities right before the collision, vector \( x_c \) consists of some auxiliary variables and impulse which rigid body and material points associated with the collision receive in Compression phase and vector \( y_c \) consists of auxiliary variables and relative velocities between rigid body and material point which are associated with the collision in the end of Compression phase. Applying Lemke’s method to Eq. (1), Eq. (1) is solved effectively by the use of the relative velocities associated with the collision at the beginning of Compression phase, that is right before the collision. Consequently, \( x_c \) and \( y_c \) are derived, in other words relative velocities and impulses associated with the end of Compression phase are derived.

Next, Extension phase is also formulated by the use of LCP as

\[ y_e = A_e x_e + b_e \]
\[ y_e \geq 0 ; x_e \geq 0 ; y_e^T x_e = 0, \]

where \( A_e \) is a matrix which consists of inertia properties and Jacobian matrix between relative and generalized velocity, \( b_e \) is a vector which includes relative velocities right after the compression phase and impulse which rigid body and material points associated with the collision receive in Compression phase, vector \( x_e \) consists of some auxiliary variables and exchanged impulse after collision which is derived based on Poisson’s law and \( y_e \) consists of auxiliary variables and relative velocities between rigid body and material point which are associated with the collision in the end of Extension phase. As well as Compression phase, applying Lemke’s method to Eq. (2), Eq. (2) is solved effectively by the use of the relative velocities and impulses derived from Eq. (1). Consequently, \( x_e \) and \( y_e \) are derived, in other
words relative velocities after the collision and impulses associated with the collision are derived.

By the use of solutions derived from two LCPs at Compression and Extension phases, velocities after collision of rigid body and material points are updated. Then, dynamics of deformable body are calculated based on MPM. In order to perform the calculation, momentums and masses of material points are projected onto the corresponding nodes by the use of interpolation function as

\[ m'_I = \sum_{p=1}^{N_p} N_I(x'_p)M_p \]

\[ p'_I = \sum_{p=1}^{N_p} N_I(x'_p)p'_p \]

where \( x'_p \) is coordinate of material point at local reference fixed to the background grid, \( N_I \) is interpolation function, \( M_p \) is the mass of the material point \( p \), \( N_p \) is the number of material points which compose the deformable bodies, \( m'_I \) is mass of node \( I \), \( p'_p \) and \( p'_I \) are the momentum of material point \( P \) and node \( I \), respectively. Then, dividing Eq. (4) by \( m'_I \), velocity of node \( I \) is obtained as

\[ v'_I = \frac{p'_I}{m'_I} \]

Using Eq. (5), gradient velocity tensor is derived as

\[ L'_p = \sum_{I=1}^{N_a} \nabla N_I(x'_p) v'_I \]

where \( N_a \) is the number of node. Then, gradient deformation tensor is updated as

\[ F^{t+\Delta t}_p = (E + L'_p\Delta t)F^t_p, \]

where \( E \) is identity matrix. Using Eq. (7), the assumed volume of material point is also updated as

\[ V^{t+\Delta t}_p = \det F^{t+\Delta t}_p V^t_p. \]

Once gradient velocity tensor is obtained, strain tensor is derived as

\[ \Delta e = (L'_p + (L'_p)^T)\Delta t/2, \]

where \( \Delta t \) is step time. Eventual goal of this study is development of analysis method for the interaction problem of dynamics between rigid body and largely deformable body including plastic deformation. However, this study is still in trial stage, hence only elastic deformation is supposed in the analysis as simple model of deformation in order to extract essential discussions on the proposed method. Then, introduction of Hooke's law yields stress increment \( \Delta \sigma^t_p \) and consequently updated stress \( \sigma^{t+\Delta t}_p \) on material point \( p \) is given by

\[ \sigma^{t+\Delta t}_p = \sigma^t_p + \Delta \sigma^t_p = \sigma^t_p + T : \Delta e, \]

where \( T \) is rigid body tensor. Using interpolation function \( N_I \), Eq. (7) and (10), derived stress on material point \( p \) is projected onto node \( I \) as
As well as derivation of elastic stress on the nodes, gravitational force on the nodes are derived by projection from the gravitational force on material point as

\[ f^\text{proj} = \sum_{p=1}^{N_j} N_j(\mathbf{x}_p^t) M_p \mathbf{g}_p^t, \]  

where \( \mathbf{g}_p^t \) is the gravitational acceleration. Adding Eq. (11) and (12), the force on node is obtained as

\[ f^\text{tot} = f^\text{elas} + f^\text{proj}. \]

If contact state is detected, that is one of the distances between the rigid body and some of material points become zero, derived force on node is projected on the material points again in order to consider the contact force. The projected force on the material points are given by

\[ f^\text{proj} = \sum_{p=1}^{N_j} N_j(\mathbf{x}_p^t) f^\text{proj}_p. \]

When contact state is detected, there are two possibilities in next time step. One is the state transition from contact state to detachment. The other is keeping the contact state. In order to determine which possibility occurs, following LCP is formulated:

\[ \mathbf{y}_x = \mathbf{A}_x \mathbf{x}_x + \mathbf{b}_x \]
\[ \mathbf{y}_x \geq \mathbf{0}; \mathbf{x}_x \geq \mathbf{0}; y^T_x \mathbf{x} = 0 \]

where \( \mathbf{A}_x \) is a matrix which consists of inertia properties and Jacobian matrix between relative and generalized velocity, \( \mathbf{b}_x \) is a vector which includes forces on material point \( f^\text{proj}_p \), vector \( \mathbf{x}_x \) consists of contact force between corresponding material points and rigid body, vector \( \mathbf{y}_x \) consists of relative acceleration between corresponding material points and rigid body. Applying Lemke’s method to Eq. (15), Eq. (15) is solved effectively and contact force \( \lambda \) is obtained. Then, contact force \( \lambda \) is projected on node as

\[ f^\text{proj}_j = \sum_{p=1}^{N_j} N_j(\mathbf{x}_p^*) \mathbf{w}_j \lambda. \]

Considering Eq. (13) and (16), force on node is derived as

\[ f^\text{tot} = f^\text{elas} + f^\text{proj} + f^\text{proj}_j \]

in the case of existence of contact. Then, positions and velocities of material points for next time step are calculated as

\[ \mathbf{v}_p^{t+\Delta t} = \mathbf{v}_p^t + \frac{\sum_{p=1}^{N_j} N_j(\mathbf{x}_p^t) f^\text{proj}_p \Delta t}{m^*_j}, \]

\[ \mathbf{X}_p^{t+\Delta t} = \mathbf{X}_p^t + \frac{\sum_{p=1}^{N_j} N_j(\mathbf{x}_p^t) \mathbf{f}_j^{\text{proj}} \Delta t}{m^*_j}. \]

where \( \mathbf{p}_i^{t+\Delta t} = \mathbf{p}_i^t + \mathbf{f}_i^{\text{proj}} \Delta t \). Additionally, dynamics of the rigid body is also obtained from following equation of motion by the use of derived \( \lambda \) as
\[ \mathbf{M} \ddot{\mathbf{q}} - \mathbf{h} - \mathbf{W} \lambda = 0, \]  

(20)

where \( \mathbf{W} \) is Jacobian matrix associated with \( \lambda \).

On the other hand, when any contact state is not detected, Eq. (18) and (19) are calculated by the use of

\[ \mathbf{f}^{\text{inj}}_i = \mathbf{f}^{\text{inj}}_i + \mathbf{f}^{\text{prj}}_i \]  

(21)

instead of Eq. (17) and consequently positions and velocities of material point of next times step is derived. Furthermore, dynamics of rigid body is given by

\[ \mathbf{M} \ddot{\mathbf{q}} - \mathbf{h} = 0. \]  

(22)

In order to study the proposed method, numerical analyzes are performed in this chapter.

### 2.3 Analyzed object and parameters of required for numerical analysis

Figure 2 shows the analyzed object which consists of a rigid body and a flexible body. In the numerical analysis, the rigid body is released from initial location and begins to fall vertically, and the rigid body collides with the flexible bodies later. As Figure 2 shows, the rigid body is located at the level of 0.015[m] over the flexible beam which both ends are fixed to rigid wall at initial state. Parameters of the system and conditions of analysis are given in Table 1.

![Figure 2](image)

**Figure 2** Analyzed object for numerical analysis

<table>
<thead>
<tr>
<th>Table 1 Parameters of the system and conditions of analysis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mass of the rigid sphere [kg]</td>
</tr>
<tr>
<td>Radius of the rigid sphere [m]</td>
</tr>
<tr>
<td>Acceleration of gravity [m/s²]</td>
</tr>
<tr>
<td>Density of flexible beam [kg/m³]</td>
</tr>
<tr>
<td>Young's modulus [Pa]</td>
</tr>
<tr>
<td>Poisson's ratio</td>
</tr>
<tr>
<td>Coefficient of restitution for normal direction</td>
</tr>
<tr>
<td>Coefficient of restitution for tangential direction</td>
</tr>
<tr>
<td>Coefficient of friction</td>
</tr>
<tr>
<td>Auxiliary variables required for impact analysis</td>
</tr>
<tr>
<td>Number of material points</td>
</tr>
</tbody>
</table>
2.4 Result of numerical Analysis

The results of numerical analysis are shown in Figure 3 and 4. In Figure 3, upper, middle and lower figures indicate the time responses of displacement, velocity and acceleration of the rigid body, respectively. In Figure 4, upper figure is the time responses of potential, kinetic and total energy of the rigid body, which depicted by dashed blue, red solid and green dot line, respectively, middle figure is the time responses of potential, kinetic, strain and total energy of the flexible beam, which depicted by dashed blue, red solid, green dot line and black dot-and-dash line, respectively, and lower figure is the time response of total energy of the system including the rigid body and the flexible beam.

According to middle figure of Figure 3 and upper figure of Figure 4, the velocity of the rigid body changes at 0.2 [s] and it loses its energy dramatically at this moment. Consequently, flexible beam also receive energy and it begins to move dramatically as middle figure of Figure 4 shows. Note that flexible beam begins to move before collision with the rigid body and the motion is caused by gravitational force because the beam has no strain in initial state. After the first collision, several collisions are observed and the motion of the rigid body converges to steady state and finally vibrational motion is observed after 1.0[s] as upper figure of Figure 3 shows. In the steady state, rigid body and flexible body have contact with each other and vibrational motion is caused by the elasticity of the flexible beam. According to lower figure of Figure 3, small magnitude of acceleration of high frequency occurs in steady state. This acceleration comes from determination procedure by use of LCP because LCP is solved in each step time and such discretization of time leads to discontinuous contact force. Furthermore, amount of total energy reduces after several collisions and it finally converges to constant value in steady state as lower figure of Figure 4 shows. This also indicates that loss of energy is caused by collision and energy loss does not happen in steady state because the rigid body and flexible beam keeps contact, i.e. no collision. Consequently, the motion observed in the numerical analysis is reasonable one in a large sense and it can be said that the proposed method shows qualitatively correct result to a certain extent.

![Figure 3 Behavior of rigid sphere in Y direction](image_url)
3 CONCLUSIONS

In this paper, a method was proposed for analysis of the system which had rigid bodies and flexible bodies with contact and collision. The proposed method consists of Pfeiffer model and MPM. A numerical example was analyzed by the proposed method and reasonable behavior was observed in the result. However, strictly qualitative and quantitative validation remains to be performed and experimental validation is also future work.

REFERENCES


CONTACT PROBLEMS USING QUATERNIONS

IN LAGRANGE EQUATIONS

RENE SOUCHET
A.F.M.
13 Rue Johann Strauss, Buxerolles,
86180, France
rp.souchet@orange.fr

ABSTRACT

In recent years, authors Udwadia et al. [1] have proposed to obtain dynamical equations using Lagrange method with generalised parameters as quaternions \( q \). In 2014, a different point of view was applied by the actual author to treat friction problems whatever the nature of the parameters e.g. quaternions. Since rigidity is not included a priori in the definition of parameters, the main aim is the necessary use of stress tensor in the Virtual Work Principle (VWP), then its elimination for rigid bodies. Here we propose to show the applicability of our method to an example involving friction expressed by inequality relations. In this paper it is highlighted the fundamental difference between constraints issued from rigidity or friction.

Key words: Lagrange equations, generalised parameters, Euler parameters.

1 INTRODUCTION

In a series of several papers[1,2,3], Udwadia et al. have published interesting papers concerning Multibody Dynamical Systems. In 2011 they resumed their previous researches and proposed a study of a spacecraft system with numerical results. The main idea of their works concerns the treatment of the mechanical equations completed by constraint equations whatever the number of parameters, even if this last number is larger than the minimum required as it is the case with quaternions. In the referenced papers, interesting introductions propose a review of earlier works on the Analytical Dynamics from the well-known Lagrange equations[4,5], but, since this subject is classical and known, it will not be repeated here.

An essential remark concerns the use of parameters. When rigidity is a priori included in the definition of parameters, as it is the case with Euler angles for rotational motions, then D’Alembert Principle gives the equations of motions by an implicit elimination of internal forces. But if rigidity is imposed by explicit relations, as in the following example describing a two-dimensional rotation by means of \((q_0,q_3)\)-quaternions, viz

\[
\begin{bmatrix}
  x_1 \\
  x_2 \\
  x_3
\end{bmatrix} =
\begin{bmatrix}
  1-2q_3^2 & -2q_0q_3 & 0 \\
  2q_0q_3 & 1-2q_3^2 & 0 \\
  0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
  X_1 \\
  X_2 \\
  X_3
\end{bmatrix}
C = [1+4q_3(q_0^2+q_3^2-1)]I_d
\]

\[ (0) \]

\( (x \) actual position; \( X \) initial position), the circle \( X^2+Y^2=1 \) become the circle \( x'^2+y'^2=1+4q_3(q_0^2+q_3^2-1) \) and this last evolution must be considered as a motion of some deformable (e.g. elastic) body if the rigidity constraint \( q^Tq=1 \) is not fulfilled. Then, if this rigidity constraint is not taken account, it is clear that such parameters require Continuum Mechanics Principles, since strains (so Cauchy stresses) must be introduced. We conclude that
D’Alembert Principles that govern rigid bodies motions only cannot be convenient for a-priori deformable bodies.

This question is fundamental. So, the present work proposes to make clear the origin of constraints, particularly if they express rigidity as in the case of quaternions or concern friction as in usual technological systems: clearly, the first one is a material constitutive law whereas the second one is a boundary condition. In the following, we highlight the origin of the mechanical equations, particularly the necessity to use the Virtual Work Principle (VWP) for a continuum with Cauchy stress tensor if rigidity is not implicitly satisfied [6]. Since we propose to illustrate the obtained results by obtaining equations in a problem involving contact with friction, we begin by a brief account of this technological constraint.

2 CONTACT WITH FRICTION

We present two equivalent forms of the Coulomb law that will be used in this paper. They are published in Duvault and Lions [7] under the two following items

\[(A) \quad |T| \leq k|N| \text{ and } |T| < k|N| \Rightarrow u_f(A) = 0, \quad |T| = k|N| \Rightarrow \exists k_i \geq 0, \quad u_f(A) = -k_iT\]

\[(B) \quad T [v_1(A) - u_i(A)] + k|N|[v_1(A)] - u_i(A)] \geq 0\]

Since the proof of this assertion is given in [7] with details, we recall briefly this proof. We note $W$ the first member of (B).

First, assertion (A) implies assertion(B).
If $|T| < k|N|$, we have $|Tv_1(A)| \leq T|v_1(A)| \leq k|N||v_1(A)|$, and, since $u_f(A) = 0$, it results

$W = T v_1(A) + k|N||v_1(A)| \geq 0$ \hspace{1cm} (2)

If $|T| = k|N|$, then $u_f(A) = -k_iT$, so that

$W = T v_1(A) + |T||v_1(A)| \geq 0$ \hspace{1cm} (3)

Second: (B) implies (A).
We first prove that $|T| \leq k|N|$. We take $\varepsilon \lambda v$, with $\varepsilon = +1 \text{ or } -1$ and $\lambda > 0$, in place of $v$ in $W$, so obtaining

$[\varepsilon Tv_1(A) + k|N||v_1(A)|] \geq \frac{1}{\lambda} [T u_i(A) + k|N||u_i(A)|]$ \hspace{1cm} (4)

But the condition $\lambda \rightarrow \infty$ implies that $[\varepsilon Tv_1(A) + k|N||v_1(A)|] \geq 0$. Then taking $\varepsilon = +1$ and $\varepsilon = -1$ gives $k|N||v_1(A)| \geq -Tv_1(A)$ and $k|N||v_1(A)| \geq Tv_1(A)$, i.e.

$|Tv_1(A)| = |T||v_1(A)| \leq k|N||v_1(A)|$ \hspace{1cm} or \hspace{1cm} $|T| \leq k|N|$ \hspace{1cm} (5)
Now we take \( v_1(A) = 0 \), so that \(-W = T u_t(A) + k |N| u_t(A) \). We examine the two cases: if \( |T| = k |N| \), then necessarily \( T u_t(A) + k |N| u_t(A) = 0 \), so that \( \exists k \geq 0 \), \( u_t(A) = -k T \), and if \( |T| < k |N| \), with \( \varepsilon = 1 \) or \(-1 \) then \( T u_t(A) + k |N| u_t(A) = (\varepsilon T + k |N|) u_t(A) \) implying that \( u_t(A) = 0 \).

### 3 Virtual Work Principle in Continuum Mechanics

It is natural to consider a single rigid body as a continuum \( B \) whose elements, called material points, constitute a three-dimensional manifold. A global configuration \( B \) (or \( B_t \) for simplicity of notations) of the body at time \( t \) is specified in an inertial coordinate frame \( T_0 = O x, y, z \) by some smooth function \( x = \chi(X, t) \), where \( x \) is the actual position of the particle located at \( X \) at initial time; \( B_0 \) is the initial configuration or some reference configuration. We note by \( O \) the point defined by \( x = \chi(O_0, t) \).

In Classical Continuum Mechanics, two geometric definitions are introduced: first the deformation gradient \( F = \partial \chi / \partial X \) that is a smooth, invertible and linear mapping, second the Cauchy-Green tensor \( C = F^T F \) that is a measure of strain in the continuous body. Naturally \( F \) and \( C \) are tensors that represent local properties around the point \( X \) at time \( t \). We note that the condition \( J = \det F > 0 \) since at initial time we have \( J_0 = 1 \). Kinematical quantities are also introduced, viz velocity \( \dot{x} = \partial \chi / \partial t \) and acceleration \( \ddot{x} = \partial^2 \chi / \partial t^2 \) on the initial configuration \( B_0 \), respectively \( u(x, t) \) and \( a(x, t) \) on the actual configuration \( B \), satisfying the identity \( \ddot{u} = (\partial F / \partial t) F^{-1} \) (note that time derivative is denoted by a point).

Contact forces developed in the interior of the body \( B \) are taken into account by the Cauchy stress symmetric tensor \( \sigma \) defined on the actual configuration \( B \). The Virtual Work Principle (in brief VWP) relies acceleration and interior forces due to Cauchy stresses to exterior forces \( f \) on the volume \( B \) and \( \varphi \) on the surface \( \Gamma \) surrounding \( B \), according satisfaction to the linear form

\[
-\int_B \rho \dot{v} dv + \int_B \rho f v dv + \int_{\Gamma} \varphi v da - \int_B \sigma : \nabla v dx = 0 \tag{6}
\]

defined on the space \( V \) of (virtual) piecewise continuous “velocities” (or “displacements”) \( v \). In the above formula, \( \rho \) designs density, the point \( (\cdot) \) is the scalar product of vectors and the double point \( (:) \) the scalar product of second order tensors. This formula is the basic statement of Classical Continuum Mechanics [6], before introduction of constitutive laws and definition of boundary and initial value problems. In a first time we apply this principle to bodies whose deformation depends on independent parameters \( q_i \) functions of time \( t \), i.e.

\[
x = G(t, q(t); X), \quad q = (q_1, q_2, \ldots, q_n) \tag{7}
\]

Finally we recall that this above principle VWP works for any part of the body and also for any system \( B = (B_1, B_2, \ldots, B_n) \) of bodies; as an example a body \( B \) may be divided into two sub-bodies.

### 4 Generalised Displacements

Now in Classical Dynamics, rigidity is a constitutive law of materials, so that the motion of such a single rigid body \( B \) is defined by some relation \( x = T_0(t) + R(t)X \) where \( T_0(t) \) is a translation and \( R(t) \) a rotation in the three dimensional space; it is well known that \( T_0 \) depends generally on three coordinates, \( R \) depends also on three coordinates and satisfies \( R^T = R \). We
note by \( O \) the point defined by \( x=T_0(t) \), translated of the origin \( X=0 \); on a practical manner we first define the point \( O \), and then the rotation \( x-T_0(t)=R(t)X \).

It is possible to consider a larger family of materials by relaxing only rigidity in the above motions, i.e. the constitutive law \( R^{-1}=R^T \). So a larger (but sufficient) family of materials is defined by requiring motions of the type \( x=T_0(t)+R(t)X \) (without \( R^{-1}=R^T \)) where \( T_0 \) and \( R \) depend on \( t \) throughout \( n+1 \) independent parameters \( q=(q_0,q_1,...,q_n)^T \). So we write for such motions of \( B \) considered in the following parts

\[
x=T_0(q(t),t)+R(q(t),t)X \quad \text{or} \quad x=T_0(q,t)+R(q,t)X
\]

So we have with notations of Continuum Mechanics

\[
F=R \quad \text{with} \quad J=\det R>0, \quad C=R^T R, \quad u=\dot{T} + \dot{R} R^{-1}(x-T_0), \quad \text{grad} u=\dot{R} R^{-1}
\]

where time derivatives are obtained from

\[
\dot{T}=(\partial T_0/\partial q_i) \dot{q}_i + (\partial T_0/\partial t), \quad \dot{R}=(\partial R/\partial q_i) \dot{q}_i + (\partial R/\partial t)
\]

Since the displacement is invertible, we have \( X=R^{-1}(x-T_0) \); so

\[
\dot{x}=\dot{T}_0 + \dot{R}X = \dot{T}_0 + \dot{R} R^{-1}(x-T_0) =
\]

\[
[(\partial T_0/\partial q_i) + (\partial R/\partial q_i) R^{-1}(x-T_0)] \dot{q}_i + [\partial T_0/\partial t + (\partial R/\partial t) R^{-1}(x-T_0)]
\]

defined on the actual configuration \( B \) (sum on repeated indices from 0 to \( n \)).

As a consequence, if constraints of rigidity are not a-priori imposed on the chosen parameters \((q_0,q_1,...,q_n)\), then necessarily we must apply the VWP to the body \( B \) under the form available in Classical Continuum Mechanics [6]. In other words, the only use of Rigid Body Dynamics Principles cannot be convenient for a-priori deformable bodies without complementary hypotheses. That is the point of view of workers in Mechanics.

**5 ELIMINATION OF CAUCHY STRESS TENSOR**

First, from the formula (6), we choose the virtual displacement fields \( v \) in \( V \) as

\[
v=G_i w_i = (T'_i + R'_i X)w_i = (T'_i + R'_i R^{-1}(x-T))w_i
\]

\[
G'_i = \partial G_i/\partial q_i, \quad T'_i = \partial T_0/\partial q_i, \quad R'_i = \partial R/\partial q_i
\]

where \( w=(w_0, w_1,...,w_n)^T \) is an arbitrary \((n+1)\)-vector. Then the usual tensor decomposition of second order leads to

\[
R'_i R^{-1}=S_i + A_i
\]

where \( S_i \) is symmetric and \( A_i \) is anti-symmetric. It results

\[
\text{grad} v=w_i R'_i R^{-1}, \quad \sigma:\text{grad} v=\sigma:w_i S_i + w_i A_i = \sigma:w_i S_i
\]
since the stress tensor $\sigma$ is symmetric and $A_i$ is anti-symmetric implying $\sigma : A_i = 0$. So, if we take the $w_i$'s satisfying the constraint $w_i S_i = 0$ (sum on $i$), then Cauchy stress tensor does not occur in the VWP written in Part 2, this fact due to the relation

$$\int_B \sigma : \text{grad} v \, dx = (\int_B \sigma \, dx) : w_i S_i$$  \hspace{1cm} (14)

As a consequence, by using these above special $w_i$'s, the various forces, by example volume forces $g$ defined on a part $B'$ of $B$ (or surface tractions on a part $\Gamma'$ of $\Gamma$) occur in the VWP by two global quantities only, the resultant $R(g) = \int_{B'} g \, dx$ and the moment $M(g) = \int_{B'} x \times g \, dx$, since we have successively

$$\int_{B'} g \cdot v \, dx = w_i \left[ (\int_{B'} g \, dx) \cdot T_i + \int_{B'} g \cdot S_i (x-T_0) \, dx + \int_{B'} g \cdot A_i (x-T_0) \, dx \right]$$  \hspace{1cm} (15)

$$= w_i T_i \cdot \left[ (\int_{B'} g \, dx) + w_i S_i : (x-T_0) \otimes g \, dx + w_i A_i \cdot \int_{B'} (x-T_0) \times g \, dx \right]$$  \hspace{1cm} (16)

$$= w_i T_i \cdot R(g) + w_i A_i \cdot (M(g) - T_0 \times R(g))$$  \hspace{1cm} (17)

where $\alpha_i = \alpha(A_i)$ is the dual vector of the anti-symmetric tensor $A_i$ (i.e. $\alpha_i \times x = A_i \cdot x$).

So we have proved the important property: in order both to remove stress tensor and to use exclusively resultant and moment of forces, it is (necessary and) sufficient to choose the arbitrary vector $w = (w_0, w_1, \ldots, w_n)^T$ satisfying the constraint

$$w_i S_i = 0 \quad \text{i.e.} \quad w_i (R_i' R^{-1})_{\text{sym}} = 0 \quad \text{(sum on $i$)}$$  \hspace{1cm} (18)

Note that in this formula, matrix $[R']$ is the derivative of the matrix $[R(q,t)]$ that was not modified by introducing some relations between parameters $q_i$'s: in other words, the parameters $q_i$ are to be treated as being independent of each other. If the only choice of parameters, i.e. without relations between them, is such that matrix $[R(q,t)]$ is a rotation, then we have $R' = R^T$, this identity implying that $R' R^T = R^T R'$ is an anti-symmetric tensor only (i.e. $S_i = 0$); so the constraint $w_i S_i = 0$ is satisfied on an identical manner and the above property exists without explicit relations. On a theoretical point of view, note that the maximum of scalar constraints equivalent to $w_i S_i = 0$ is six since $S_i$ is symmetric.

Second, we explicit the VWP by choosing arbitrary vector $v$ such that the constraint $w_i S_i = 0$ is fulfilled. From

$$v = (T_i', + R_i' (x-T_0)) w_i \quad , \quad w_i S_i = 0$$  \hspace{1cm} (19)

the VWP takes the form

$$-L_i w_i + [R(f) + R(\phi)] w_i T_i' + [M(f) - T_0 \times R(f) + M(\phi) - T_0 \times R(\phi)] w_i \alpha_i = 0$$  \hspace{1cm} (20)

$L_i$ being the well known term of Lagrange

$$L_i = \int_{B_i} \rho a \cdot G_i \, dX = (d \frac{\partial}{\partial t} \frac{\partial}{\partial q_i}) K$$  \hspace{1cm} (21)
where $K$ is the kinetic energy. We highlight that these last equations are consequences of the VWP only, whatever the material of the body (suffering motions (3)), under the only constraint $w_i S_i = 0$.

What arrives if the above constraint is not imposed on parameters? In this case the $(n+1)$ Lagrange equations follow

$$-L_i + \int_B \rho f \cdot G_i' \, dx + \int_\Sigma \varphi G_i' \, da - \int_B \sigma dx : S = 0 \quad (22)$$

i.e.

$$[(-L_i + (R(f) + R(\varphi)) T_i' + (M(f) - T_0 \times R(f) + M(\varphi) - T_0 \times R(\varphi)) \cdot \alpha_i]$$

$$+ [(\int_B \rho \varphi \otimes (x - T_0) \, dx + \int_\Sigma \varphi \otimes (x - T_0) \, da - \int_B \sigma dx) : S_i] = 0 \quad (23)$$

First, if parameters are such that $R$ is a rotation, then $S_i = 0$ and we obtained the usual Lagrange equations. Second, if $R$ is not a rotation, then it is seen that the second bracket containing Cauchy stress tensor is present in this relation. So the VWP cannot eliminate this expression of internal actions. Also the Lagrange equations given in references [1,2,3] contain such a term in the right member and not the only generalized “given” forces and torques acting on the body.

6 CONSTITUTIVE LAW OF RIGIDITY

We recall that $w_i S_i = 0$ (sum on $i$) is a constraint that permits not only to eliminate Cauchy stress tensor but also to take account of global efforts only. However, this constraint does not concern the material that constitutes the body. An experimental property used in Analytical Dynamics is the rigidity of the material, that is a constitutive law explicitly written under the form

$$C = Id \quad \text{or} \quad R^i = R^T \quad (24)$$

By example, the constraint $q_1^2 + q_2^2 + q_3^2 = 1$, or $q^T q = 1$, assures rigidity if Euler parameters are used. Also this relation simplifies the calculus of the matrix $R^i R^{-1} = S_i + A$, since avoiding the calculus of the inverse matrix and simplifying the expressions of $w_i S_i$ and $w_i \alpha_i$. We recall that the expression of $R^i$ must be obtained with independent parameters in order to use Lagrange formulae (21).

Finally, a dynamical problem is to formulate by taking account of others bodies so constituting some system $S$ with constraints between bodies, such punctual friction by example or other various technological constraints. In the next part we give an example of contact with friction.

7 CONTACT WITH FRICTION

We consider an homogeneous rigid wheel (centre $O$, radius $r$ and mass $m$) rolling in a vertical plane $O_0x_0y_0$ on an inclined line (or surface) $O_0x_0$ under the gravitational acceleration $g$ downwards, the gravitational force being $f = -mgv_0$ applied on the centre $O$ of the wheel. We use the referential $Ref = O_0x_0y_0z_0$ with the angle between $O_0x_0$ and $O_0x_0$ noted $a$. Three parameters define the configurations of the wheel in the referential $Ref$: position $(x,y)$ of the centre $O$ and rotation $\vartheta$ of the wheel. So the parameters $(x,y, \vartheta)$ preserve the rigidity of the body, implying that the matrices $S_i$ are identically null. A classical treatment may be used to solve this problem.
A- But in this part two-dimensional Euler parameters \((p,q)\) are introduced to specify the rotation of the wheel, so writing

\[
R = \begin{bmatrix} 1-2q^2 & -2pq \\ 2pq & 1-2p^2 \end{bmatrix}, \quad R^t = R^t/\Delta, \quad \Delta = 1+4q^2(p^2+q^2-1) \tag{25}
\]

with \(\Delta = 1\) if \((p^2+q^2=1)\). We highlight that the condition \(\Delta = 1\) expresses a rigidity condition \((R^t=R^t)\) for the wheel, so implying that this relation is not to be introduced in the Lagrange terms of the VWP, but may be used in the others terms of this principle.

Now we introduce the virtual coefficients \((w_x,w_y,w_p,w_q)\) associate to the parameters \((x,y,p,q)\) and the condition \(w_iS_i=0\), i.e. (after some calculus)

\[
4pq^2w_p+\{4p^2q-4q(1-2q^2)\}w_q=0 \quad \text{or} \quad pw_p+qw_q=0 \tag{26}
\]

(taking account of \(p^2+q^2=1\)), so that the virtual displacement of the point \(x\) of \(B\) is

\[
v(x) = (v_1(x),v_2(x),0) = (w_x,w_y,0)+(\alpha_x w_p+\alpha_q w_q)z_0 \times (x-T_0) \tag{27}
\]

\[
\alpha_p = 2q(1-2q^2), \quad \alpha_q = 2p(1+2q^2)
\]

\[
v(0) = (w_x,w_y,0), \quad v(A) = (v_1(A),v_2(A),0) = (w_x+r(\alpha_p w_p+\alpha_q w_q),w_y,0) \tag{28}
\]

In this formula \(A\) is the contact point of \(B\) with the axis \(O_0X_0\) and its velocity is given by

\[
u(A)= (\dot{x}+(r/\sqrt{\Delta}))[2q(1-2q^2)p+2p(1+2q^2)\dot{q}], \quad \dot{y},0 \tag{29}
\]

Under the above conditions, the VWP is written

\[
\int_B \rho a_v \, dx \cdot mgy_0 \cdot v(O)+Tv_1(A)+Nv_2(A)=0 \tag{30}
\]

where \((T,N,0)\) are the components of the two-dimensional contact force on the wheel applied at the contact point \(A\). The integral term is the virtual work of acceleration that may be obtained from the kinetic energy \(K\) of the wheel by means of Lagrange formula, with

\[
K = \frac{1}{2}m(\dot{x}^2+\dot{y}^2)+mr^2[4q^2\dot{q}^2+(\dot{p}q+p\dot{q})^2] \tag{31}
\]

this last expression being obtained (after some calculus) with independent parameters \((x,y,p,q)\).

The expression of VWP contains the two components of the contact force. Now we must use the contact law of friction, by example in the hypothesis of a bilateral contact \((y=r)\) at the point \(A=(x,y-r,0)\) of the wheel, implying the geometric constraint \(y=r\), together with the Coulomb law of friction:

\[
\lfloor T \rfloor \leq k \lfloor N \rfloor \quad \text{and} \quad \lfloor T \rfloor < k \lfloor N \rfloor \Rightarrow u_t(A)=0, \quad \lfloor T \rfloor = k \lfloor N \rfloor \Rightarrow \exists k_i \geq 0, \quad u_t(A)=-k_i T \tag{32}
\]

where \(k\) is a (constant) friction coefficient and \(u_t(A)\) the sliding velocity. We have proven that this law is equivalent to the inequality [7]
So we rewrite the VWP with \((v-u)\) in place of \(v\). It follows

\[
\int_B \rho a. (v-u) dx + mg y_0 \left[ v(O)-u(O) \right] - N \left[ v_2(A)-u_2(A) \right] = T[ v_1(A)-u_1(A) ]
\]

(34)

Using this expression of \(T[ v_1(A)-u_1(A) ]\) into the formula (33) leads to the inequality

\[
\int_B \rho a. v dx + mg y_0 v(O)-N v_2(A)+ k \left[ N \| v_1(A) \| \right] \geq \int_B \rho a. u dx + mg y_0 u(O)-N u_2(A)+ k \left[ N \| u_1(A) \| \right]
\]

(35)

Now we note (conservation of mass) that

\[
\int_B \rho a. u dx = \dot{K}
\]

(36)

so that the above inequality (35) becomes

\[
\int_B \rho a. v dx + mg y_0 v(O)-N v_2(A)+ k \left[ N \| v_1(A) \| \right] \geq \dot{K} + mg y_0 u(O)-N u_2(A)+ k \left[ N \| u_1(A) \| \right]
\]

(37)

We observe that the first member is a linear form on the virtual velocity space whereas the second member is constant on this space. In the next part this remark will be used to obtain essentials properties.

B. Now parameters are specified such that \(w_x=w_p=w_q=0\) , satisfying \((w_iS_i=0)\). It results \(v(x)=(0,wy,0)\) so that by taking account of the bilateral contact \(y=r\)

\[
(mgcosa-N)w_y \geq \dot{K} + mgsina \dot{x} + k \left[ N \| u_1(A) \| \right]
\]

(38)

This inequality is satisfied whatever the arbitrary coefficient \(w_y\) if and only if

\[
(mgcosa-N)=0 \text{ and } \dot{K} + mgsina \dot{x} + k \left[ N \| u_1(A) \| \right] = 0
\]

(39)

If the first equality has an evident origin \((y=r)\), we remark that the second equality expresses the kinetic energy theorem. In fact the real power of internal forces is null under the constraint \(w_iS_i=0\) and the real power of the contact force is given by the scalar product \(T u_1(A)+N u_2(A)\) + \(T u_1(A)\) = \(-k \left[ N \| u_1(A) \| \right]\) by taking account of the Coulomb friction law. So it results from (37)

\[
\int_B \rho a. v dx + mg y_0 v(O)-N v_2(A)+ k \left[ N \| v_1(A) \| \right] \geq 0 \quad \text{where } N=mg cosa
\]

(40)

that is available whatever the parameters \((w_x,w_p,w_q)\).

After some straightforward calculus, the term \(\int_B \rho a. v dx\) is obtained under the form

\[
\int_B \rho a. v dx = m\ddot{x}w_x + 2m r^2 (Aw_y+Bw_q)
\]
Also we can write
\( A = q^2 \ddot{p} + pq \dot{q}^2 + 2q \dot{p} \dot{q} \) \quad (41)

\[ B = \dot{p} q \dot{q} + p^2 \dot{q}^2 + 4q^2 \dot{q}^2 + 2pq \dot{q} + 4q \dot{q}^2 \]

Taking account of this expression, the differential variational inequality follows
\[ (\dot{m} + mgsina)w_1 + kmg cosa |w_1 + rw_3| \geq 0 \] 
under the compatibility condition \( pw_2 + qw_2 = 0 \). That is the basic relation to solve the problem completed naturally by initial conditions on velocities (and positions). The numerical treatment of this inequality is not the aim of this present mechanical work.

C. If the parameters \((x,y,\vartheta)\) preserving the rigidity of the wheel are chosen, then the following variational inequality is obtained from (40)
\[ (m \ddot{x} + mg sina)w_1 + l \dot{\vartheta} w_3 + k mg cosa |w_1 + rw_3| \geq 0 \] 
whatever the virtual displacement \((w_1, w_3)\).

Note: It is easy to explicit this relation. After some calculus we obtain
\[ (m \ddot{x} + m \ddot{y}) w_1 + l \dot{\vartheta} w_3 + K + mg(w_1 \dot{x} + sina + mg(w_2 \dot{y} cosa) \\
+ k |N||w_1 + rw_3||\dot{x} + r \dot{\vartheta}| - N(w_2 \dot{y}) \geq 0 \] 

But we must take account of the geometric contact \( y=r \) so that
\[ (m \ddot{x} + mg sina)w_1 + l \dot{\vartheta} w_3 + (mg cosa \cdot N) w_2 + k |N||w_1 + rw_3| \geq \\
K + mg \dot{x} sina + k |N||\dot{x} + r \dot{\vartheta}| \] 

This inequality must be satisfied whatever the virtual displacement \((w_1, w_2, w_3)\). We can choose \( w_1=w_3=0 \), so that whatever \( w_2 \)
\[ (mg cosa \cdot N) w_2 \geq K + mg \dot{x} sina + k |N||\dot{x} + r \dot{\vartheta}| \] 

But this inequality is also satisfied by \((-w_2)\) and that is possible if and only if
\[ N=mg cosa \quad and \quad K + mg \dot{x} sina + k |N||\dot{x} + r \dot{\vartheta}| = 0 \] 

This second equality is the kinetic energy theorem applied to the wheel. So we have the variational differential inequality
\[ (m \ddot{x} + mg sina)w_1 + l \dot{\vartheta} w_3 + k mg cosa |w_1 + rw_3| \geq 0 \] 
whatever the virtual displacement \((w_1, w_3)\).
8 CONCLUSION

The present paper has presented a natural link existing between Analytical Dynamics and Continuum Mechanics. The key of the actual scheme was the use of the Virtual Work Principal; then the elimination of Cauchy stresses introduces compatibility relations between virtual coefficients [8,9]. The example on friction has highlighted the necessity to separate these compatibility conditions and the relations expressing mechanical laws.

REFERENCES

Damping Amplification Caused by a Mechanism that Trans-pass Through its Singular Position

Krzysztof Lipinski

Mechanical Department
Gdansk University of Technology
ul Narutowicza 11/12, 80-299, Gdansk, Poland
klipinsk@pg.gda.pl

ABSTRACT
In the paper, vibrations of a hybrid multibody-continuous system are investigated. Despite of their presence in the every day experiences, vibrations are undesired in most of the technical cases. In the paper, a damping method based on modal disparity is investigated. Low frequency vibrations should be damped. To accelerate it, energy transfer between different modes is proposed. This idea is tested numerically. Considered model consists of two parts: an elastic beam part and a multibody part. Beam is the main vibrating element. The multibody part is set in a configuration close to its singular position. For higher ranges of motion, extension of the mechanism become impossible and a point of the beam becomes locked. As a result, variation of the shape of the vibration modes is observed and energy is transferred to some of the higher frequency modes, and dissipated. Presented numerical tests are performed with different values of the structural damping, as well as with different lengths of the mechanism’s arms.

Keywords: Vibrations, Damping, Modal disparity, Finite elements, Constraint equations.

1 INTRODUCTION
In the paper, an effective damping method is searched and vibrations of a hybrid multibody-continuous system (Fig. 1) are investigated. Despite of their presence in the every day practice, vibrations are undesired in most of mechanical constructions. Except of some exceptional cases (when vibrations are generated intentionally), vibrations have to be eliminated. Industrial significance of the problem is substantial. A short overview of their negative outcomes can be cited. The uncontrolled motion of the effectors is one of the main outcomes. With the effector vibrations, the product required quality can be difficult to achieve [5, 6]. The straightforward elasto-inertial effects are not the only responsible for the elements’ vibrations. Vibrations at the sensors surrounding can indicate a noised signal introduced in the control loop [1] and the manual (or automatic) control of the system can amplify these vibrations, as inertia of the operator’s hands can be too high to balance vibrations of the control bars. Significant acoustic noises have to be pointed. Uncontrolled motions (vibrations) of the surrounding elements can be evoked, when synchronized in frequencies. Some long-term effects have to be pointed, too. Harmful influence on the human health [2, 3, 4, 7, 8], and the fatigue effects on the mechanical elements are the examples. Summarising it, a search for an effective damping method is one of the crucial topics in the most of the industrial design processes.

Methods based on the viscous dampers look as the dominant solution in the presently used damping methods. Successfully applied in the medium size applications (as vehicles for example), they are difficult to operate in smaller size applications, where the electromagnetic
effects are challenging [9, 10]. Electromagnetic elements reinforce their positions thanks of their lower dimensions, simpler constructions, easier semi active control and luck of the potential oil impurities.

At one of the paper’s initial presumption, we have considered that some quantity of internal structural damping is present in most of the industrial applications. According to it, high frequency vibrations (except of the self-excited or the resonance cases) are damped well. External action is not necessary. Harmonic vibrations of low frequency are more challenging, as intensity of the structural damping depends on the velocity of element deformation, and the last is proportional to the frequency. To accelerate the damping, external velocity amplification is useful [11, 13, 14], and energy transfer (modal disparity) between different modes is helpful [12]. In the paper, the modal disparity is investigated. We focus on a double-bar mechanism introduced between the vibrating element (the vibrating beam) and the motionless reference (Fig. 1). Singular position of the double-bar is crucial for the investigated behaviour. Like in a case of the unilateral constraints, depending on sense of the motion, number of the system degrees of freedom is varying. According to it, different modes of vibrations are present depending on the sense of the motion. Energy transfer between the modes is unavoidable for the singular position trans-passing. Strong nonlinearity of the system in the neighbourhood of the singular position is useful, too.

Six sections are proposed in the paper. For the title announced mechanism (a double-bar mechanism), a multibody model is proposed. Fundaments of the used multibody formalism are presented in section two. To prepare the model, the classical (rigid body) multibody modelling is considered. The system bodies are coupled one to the other by use of massless joints. To obtain dynamics equations of the system, a method presented in [7, 8] is employed. Three-like system is considered. Joint relative coordinates are used as the system generalised coordinates.

To deal with the elastic beam structure, finite elements method is used and presented in section three. A method proposed in [10-12] is recalled. The considered elastic structure is modelled as composed of short deformable elements, i.e. the finite elements. Displacements of nodes are taken as system’s generalized coordinates (the set of nodes is understood as a finite set of some initially selected cross sections of the considered beam). Small values of the node displacements are considered, and the overall drift motion of the elastic structure is absent. Obtained dynamic equations are linear (in respect to the joint displacements, velocities and accelerations). Related formulas for the mass, dissipation and stiffness matrices are detailed in the third section.

To joint the two structurally different sub-models, constraint equations are proposed for the contact between the mechanism and the vibrating element. For the obtained closed-loop system, a Lagrange multipliers technique is used to describe its dynamics. To eliminate the Lagrange’s multipliers, as well as the dynamics equations related to the dependent accelerations, a slightly modified elimination technique is proposed. It is based on the classical coordinate partitioning method is used [15]. Details of the constraint equations, as well as the elimination method, are presented in section four. Details of the proposed models (numerical data), as well as the obtained results are presented in section five. Finally, conclusions and perspectives are presented in section six.

2 FUNDAMENTS OF THE USED MULTIBODY FORMALISM

In the considered multibody modelling, equations are based on joint coordinates as the system generalized coordinates. The multibody sub-system is composed of rigid bodies. Massless joints are used to connect them. Magnitudes of the joint displacements are considered as significant. In the system’s reference “zero” position, the joint axes are collinear with the axes of the inertial motionless reference coordinate system for all of the joints considered in the system.

To unify the nomenclature: a name material body is associated to a body of finite size and inertia; the reference body is motionless and infinitely inertial; massless bodies are finite in dimensions but with negligible inertia parameters; point bodies are massless and dimensionless.
Joints are one-degree-of-freedom (restricted to prismatic or revolute type). Any of the multi-degrees-of-freedom connections is modelled as a short kinematical chain composed of joints, point bodies and eventually constraints. Kinematical chain is a sequence of bodies connected by joints (Fig. 2a). Reference kinematical chain of body $i$ is the chain that connecting the generic body $i$ with the reference body. Open kinematical chain is a chain where the reference kinematical chain is defined uniquely for any of its bodies. Otherwise, closed kinematical chain is formed (Fig. 2a). When none on the system chains is of the closed type, the system is called as a tree structure (Fig. 2b). Numbering can be introduced for bodies of tree structure systems (Fig. 2b). The numbering coincides with the succession order present in the reference chain. Then $a < b$ symbol is used when $a$ belongs to the reference chain of the generic body $b$. Symbol $a$ denotes the complete set of direct successors and symbol $a^+$ denotes the direct predecessor of body $a$. Finally, a joint is numbered as $a$ what it is located between body $a$ with the body $a^+$.

Figure 2: Examples of the multibody systems: closed loop structure (a); body numbering (b); main distances and interactions at the generic body $i$ of the multibody system (c).

To describe geometrical properties of the generic body $i$, we use vectors presented in Fig. 2c. Their coordinates have to be recalculated from the body fixed to the reference fixed coordinate system. Related formulas depend on joint orientation matrices $T^j$ calculated as some ordered products of the relative orientation matrices $R^j$, i.e., products of orientation changes in joints of the reference chain of the body $i$. To calculate vector $x^i$ (it locates the mass centre of body $i$ in respect to the origin of the reference system) all translations and body dimensions of the reference chain of the generic body $i$ are summed. It leads to [13, 16, 17]:

$$ T^i = \prod_{j \in i} R^j; \quad x^i = \sum_{j \in i} \left( \frac{\bar{x}^j}{R_i} + \bar{d}^j \right). \quad (1) $$

When the orientation matrices and the position vectors are differentiated in respect to time, and when a matrix form is used, ones obtain [13, 16, 17]:

$$ \dot{\omega}^j = \Delta^{1j} \cdot \dot{q}^b; \quad \ddot{x}^i = \Delta^{1i} \cdot q^b; \quad (2) $$

$$ \dot{\omega} = \Delta^{1i} \cdot \dot{q}^b + \vec{\omega}^R; \quad \ddot{x} = \Delta^{1i} \cdot \ddot{q}^b + \dddot{x}^R, \quad (3) $$

where: $q^b$ - column matrix of system generalized coordinates; $\Delta^{1j}, \Delta^{2j}$ - row matrices of partial vectors; $\vec{\omega}^R, \dddot{x}^R$ - “remainders” independent of joint accelerations.

To obtain dynamics equations, free body diagrams are composed (Fig. 2c). All joints are cut in the multibody structure and joint interactions are introduced to replace the joints. Then, the Newton/Euler equations of dynamics are used [13, 16, 17]:

$$ m^i \cdot \dddot{x}^i = \bar{f}_{i\in c} + \bar{f}_{i\in c} - \sum_{j \in i} \bar{f}_{j}^i; \quad (4) $$

$$ \vec{\omega} \times (\bar{T}^i \cdot \dot{\vec{\omega}}) + \bar{T}^i \cdot \dot{\vec{\omega}} = \bar{r}_{i\in c} + \bar{r}_{i\in c} - \sum_{j \in i} \bar{r}_{j}^i - \sum_{j \in i} \bar{r}_{j}^i \times \bar{f}_{j}^i, \quad (5) $$

where: $m^i$ - mass of the $i^{th}$ body; $\bar{T}^i$ - $i^{th}$ body’s tensor of moments of inertia about the mass centre; $\bar{f}_{i\in c}, \bar{r}_{i\in c}$ - force and torque at the $i^{th}$ joint; $\bar{f}_{j}^i, \bar{r}_{j}^i$ - generic body’s net external force at the mass centre; $\bar{r}_{j}^i \times \bar{f}_{j}^i$ - $i^{th}$ body’s net external torque about the mass centre.

1 All elements of these matrices are geometrical vectors

---

522
Joining together the dynamics eqs (4)-(5) with the kinematics eq. (3), one obtains \[13, 16, 17\]

\[
\dot{\mathbf{B}}^{1i} \cdot \mathbf{q}^b + m^i \cdot \ddot{x}^{i,i} = \ddot{f}_i + \dot{f}_j = -\sum_{j \in I} \dot{f}_j^i;
\]

\[
\dot{\mathbf{B}}^{2i} \cdot \mathbf{q}^b + \dot{\omega} \times \left[ \ddot{\mathbf{T}} \cdot \dot{\omega} \right] + \ddot{\mathbf{T}} \cdot \dot{\omega} \times \mathbf{R} = \ddot{t}_{ic} + \dddot{t}_{jc} + \dot{t}_{ic} - \sum_{j \in I} \dot{t}_{jc}^i - \sum_{j \in I} \dddot{t}_{ic}^j \times \ddot{f}_j^i,
\]

where: \( \mathbf{B}^{1i} = m^i \cdot \mathbf{A}^{1i} \), \( \mathbf{B}^{2i} = \ddot{\mathbf{T}} \cdot \mathbf{A}^{2i} \) - matrices of the partial vectors.

The first summands are not the lonely terms dependent on the joint accelerations in eqs (6) and (7). Forces and torques of the successors’ joints are the others. To obtain the explicit formula, backward evaluation is employed. The process starts from leaf bodies (i.e., these that haven’t any successors). At each of the iterative step, the \( i^\text{th} \) interactions are replaced by the formulas obtained in the previous step. After it, the \( p^\text{th} \) iteration, the \( i^\text{th} \) joint interactions are the only present in the dynamics equations. Placing these interactions at the left side of the dynamic equation, the matrix form of this equation is \[13, 16, 17\]:

\[
\dddot{\mathbf{B}}^i = \mathbf{C}^{L,i} \cdot \mathbf{q}^b + \mathbf{D}^{L,i} + \mathbf{E}^{L,i};
\]

\[
\dddot{\mathbf{B}}^i = \mathbf{C}^{2,i} \cdot \mathbf{q}^b + \mathbf{D}^{2,i} + \mathbf{E}^{2,i},
\]

where:

\[
\mathbf{C}^{L,i} = \sum_{j \in I} \dot{f}_j^i;
\]

\[
\mathbf{C}^{2,i} = \sum_{j \in I} \left[ (\mathbf{B}^{2j} + \sum_{d \in \text{I}} \dddot{t}_{ic}^d) \times \mathbf{B}^{1j} \right];
\]

\[
\mathbf{D}^{L,i} = \sum_{j \in I} \dddot{t}_{ic}^j ;
\]

\[
\mathbf{D}^{2,i} = \sum_{j \in I} \left[ \dddot{t}_{ic} + \dddot{t}_{jc} + \dddot{t}_{ic} - \sum_{j \in I} \dot{t}_{jc}^i - \sum_{j \in I} \dddot{t}_{ic}^j \times \ddot{f}_j^i \right];
\]

\[
\mathbf{E}^{L,i} = -\sum_{j \in I} \dddot{t}_{ic}^j ;
\]

\[
\mathbf{E}^{2,i} = -\sum_{j \in I} \dddot{t}_{ic}^j - \sum_{j \in I} \dddot{t}_{ic}^j \times \ddot{f}_j^i.
\]

Interactions (8) are projected onto joint mobility vectors (\( \mathbf{c}^i \) for rotational and \( \mathbf{a}^i \) for translational). It is supposed, the \( \mathbf{c}^i \) vector is zero for the translational joint, and the \( \mathbf{a}^i \) vector is zero for the rotational joint. It turns the dynamics equations into a matrix from [13, 16, 17]

\[
\mathbf{M}^b(\mathbf{q}^b) \cdot \ddot{\mathbf{q}}^b + \mathbf{F}^b(\mathbf{q}^b, \mathbf{q}^b) + \mathbf{Q}^b(\dot{\mathbf{q}}^b, \mathbf{q}^b, \mathbf{f}_e, \mathbf{t}_e, \mathbf{t}) = 0,
\]

where:

\[
\mathbf{M}^b = \mathbf{a}^i \circ \mathbf{C}^{1,i} + \mathbf{c}^i \circ \mathbf{C}^{2,i};
\]

\[
\mathbf{F}^b = \mathbf{a}^i \circ \mathbf{D}^{1,i} + \mathbf{c}^i \circ \mathbf{D}^{2,i};
\]

\[
\mathbf{Q}^b = \mathbf{a}^i \circ \mathbf{E}^{1,i} + \mathbf{c}^i \circ \mathbf{E}^{2,i}.
\]

3 FUNDAMENTS OF THE USED FINITE ELEMENTS METHOD

A continuous one-dimensional deformable system is present in the considered mechanical construction. To model this system, finite elements are used [18, 19]. A finite set of points is distributed along the neutral line of the modelled beam. They are called nodes. The inertial and elastic space bounded by the two neighbour cross sections is called the finite element. Displacements of the cross sections are considered as the generalized coordinates of the beam. It is supposed, the cross sections remain planar and do not change their shapes and areas during deformations. In the general case, six degrees of freedom are necessary to describe displacement of each of the cross sections. (i.e. three of them are the translational displacements of the node and the three other are rotation angles of the section). In the modelled case, when a planar beam system is considered (Fig. 3a), two degrees of freedom are sufficient (a vertical translation and an in-plane rotation). The longitudinal displacement of the node is considered is infinitesimally small in compare to the two previous.

![Figure 3: Details of the considered finite element: displacements for a beam element (a); blocks of inertia matrix in local set coordinates (b); blocks of inertia matrix in global set coordinates (c) ][figure3]

[523]
Next, two sets of coordinate systems are introduced: a global coordinate system is fixed to the reference body and local coordinate systems coincide with localisation of non-deformed elements. Parameters expressed in the local coordinate system are marked with “hats” \( ^\hat{\cdot} \) (located above of the parameter). By a convention, the vertical displacements are performed along the \( y_2 \) axis, while the rotations are performed about the \( y_3 \) axis. When the two nodes of the element are numbered as \( i \) and \( j \), the elements coordinates are

\[
\hat{\mathbf{q}}_e = \text{col}(\hat{q}_{i2}, \hat{q}_{i6}, \hat{q}_{j2}, \hat{q}_{j6}) ,
\]

where: \( \hat{q}_{i2}, \hat{q}_{j2} \) - translations of the beam nodes, \( \hat{q}_{i6}, \hat{q}_{j6} \) - rotations of the beam cross sections

Loads (when present in the system) are distributed over all the particles of the elements. When reduced to the nodes, they are denoted as

\[
\hat{\mathbf{P}}_e = \text{col}(\hat{P}_{i2}, \hat{P}_{i6}, \hat{P}_{j2}, \hat{P}_{j6}) ,
\]

where: \( \hat{P}_{i2}, \hat{P}_{j2} \) - forces collinear to \( y_2 \); \( \hat{P}_{i6}, \hat{P}_{j6} \) - torques collinear to \( y_3 \).

When a generic point \( A \) is located at the neutral line of the finite element (Fig. 3a), it displacement, \( \hat{\mathbf{q}}_A \), is approximated with use of a linear function of displacements, \( \hat{\mathbf{q}}_e \), of the element nodes. It can be expressed in a matrix form as

\[
\hat{\mathbf{q}}_A = \hat{\mathbf{N}}_e \cdot \hat{\mathbf{q}}_e ,
\]

where: \( \hat{\mathbf{q}}_A \) - vector of displacements of the generic point \( A \); \( \hat{\mathbf{N}}_e = \text{col}(N_1, N_2, \ldots, N_n) \) - matrix of shape functions (nonlinear functions of \( l \)); \( \hat{\mathbf{q}}_e \) - vector of the nodes displacements of eq. (10).

Equation (12) is linear in respect to \( \hat{\mathbf{q}}_e \), but the coefficients (elements of the \( \hat{\mathbf{N}}_e \) matrix) nonlinearly depend on the relative position of point \( A \), \( \hat{x}_{3A} \), measured in respect to the nodes. Proposed coefficients of \( \hat{\mathbf{N}}_e \) should preserve continuity conditions at elements’ connecting points. In the general case, nonlinear polynomials are used to express these coefficients as the functions of \( \hat{x}_{3A} \). Usually, cubic functions are used as the shape functions of beam elements:

\[
\hat{\mathbf{N}}_e(\zeta) = \begin{bmatrix} 2\zeta^3 - 3\zeta^2 + 1 & l_e (\zeta^3 - 2\zeta^2 + \zeta) & -2\zeta^3 + 3\zeta^2 & l_e (\zeta^3 - \zeta^2) \\ \frac{6(\zeta^2 - \zeta)}{l_e} & 3\zeta^2 - 4\zeta + 1 & \frac{6(-\zeta^2 + \zeta)}{l_e} & 3\zeta^2 - 2\zeta \end{bmatrix} ,
\]

where: \( \zeta = \hat{x}_{3A}/l_e \) - point \( A \) relative position; \( l_e \) - length of the considered element (see Fig. 3a).

As displacements of point \( A \) are approximated, approximation errors are present. To minimize them, elements size should be relatively small.

With the approximated functions of the displacements of the generic point \( A \), the element’s relative strains, \( \varepsilon_A \), can be got from a differential relation

\[
\varepsilon_A = \hat{\mathbf{B}}_I \cdot \hat{\mathbf{q}}_e : \hat{\mathbf{B}}_I = \begin{bmatrix} \partial / \partial s_1 & 0 & 0 & \partial / \partial s_3 \\ 0 & \partial / \partial s_5 & 0 & \partial / \partial s_3 \\ 0 & 0 & \partial / \partial s_5 & \partial / \partial s_3 \end{bmatrix} \cdot \hat{\mathbf{N}}_e ,
\]

where: \( \hat{\mathbf{B}}_I \) - matrix that relates displacements and the element’s strains.

As in the considered case, a planar state of stresses is analysed, the stress/strain relation is

\[
\hat{\sigma} = \hat{\mathbf{D}}_e \cdot \varepsilon = \frac{E}{1 - \nu} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & (1-\nu)/2 \end{bmatrix} \cdot \hat{\mathbf{N}}_e .
\]
Next, the total kinetic and potential energies of the considered finite elements are calculated from integrals:

\[ E_k = \frac{1}{2} \int \hat{q}^T \hat{q} \; dm \quad \text{and} \quad \int \hat{q}^T \cdot \hat{q} \; dA, \]  

when Eqs. (12) (14) and (15) are used, these equations can be rewritten as [13, 18, 19]:

\[ E_k = \frac{1}{2} \hat{q}_e^T \cdot \hat{A}_e \cdot \hat{q}_e \quad \text{and} \quad V_e = \frac{1}{2} \hat{q}_e^T \cdot \hat{C}_e \cdot \hat{q}_e, \]  

where:

\[ \hat{A}_e = \rho \cdot \int \int \int \hat{N}_e^T \cdot \hat{N}_e \cdot dx \cdot dy \cdot dz \quad \text{and} \quad \hat{C}_e = \int \int \int \hat{B}_e^T \cdot \hat{D}_e \cdot \hat{B}_e \cdot dx \cdot dy \cdot dz. \]  

When integrated, the mass and the elasticity matrices are [13, 18, 19]:

\[ \hat{A}_e \begin{bmatrix} 156 & 22l_e & 54 & -13l_e \\ 24l_e^2 & 13l_e & -3l_e^2 & 156 \\ -22l_e & 13l_e & 24l_e^2 & -3l_e^2 \\ -4l_e^2 & -3l_e^2 & 4l_e^2 & -4l_e^2 \end{bmatrix}_{\text{sym.}} \quad \text{and} \quad \hat{C}_e = \frac{EJ_e}{l_e} \begin{bmatrix} 12 & 6l_e & -12 & 6l_e \\ 6l_e & 12 & -6l_e & 6l_e \\ -12 & -6l_e & 12 & -6l_e \\ 6l_e & 6l_e & -6l_e & 12 \end{bmatrix}_{\text{sym.}}. \]  

Finally, the damping matrix is approximated as a linear combination of the mass and the elasticity matrices [13, 18, 19]

\[ \hat{B}_e = \alpha \hat{A}_e + \beta \hat{C}_e. \]  

As in the presently considered system, the local coordinates and the global coordinates are oriented identically (none of additional rotations is necessary), the introduced local matrices are used in the global system directly. To join the set of the independent elements into the common construction, vectors of the nodes’ displacements are partitioned on two sub-vectors of each of the nodes considered separately (a similar action is performed for loads, too). Then, the sub-vectors are collected into a common matrix \( \hat{q}^* = \text{col}(q_i) \), where \( q_i \) is a sub-vector composed of displacements of the \( i^{th} \) node. To obtain the corresponding matrices, \( \hat{A}^*, \hat{C}^*, \hat{B}^* \), of the element dynamics expressed with use of the completed system of the generalized coordinates, matrices (19) and (20) are partitioned on four blocks each. Rows and columns with numbers matching to the positions of displacements of the given node are collected together in the corresponding sub-blocks (Fig. 3b). The sub-blocks of the local \( \hat{A}_e, \hat{C}_e, \hat{B}_e \) matrices are placed in related places of the global matrices \( \hat{A}_e, \hat{C}_e, \hat{B}_e \) (Fig. 3c). The other elements of the \( \hat{A}_{\text{ex}}, \hat{C}_{\text{ex}}, \hat{B}_{\text{ex}} \) matrices are kept to be zero. The final global matrices of the complete system are obtained as sums of the elements’ global matrices (summation is performed over all the elements of the considered continuous system) [13, 18, 19]:

\[ A^* = \sum_{i=1}^{n_e} A_i^* \quad \text{and} \quad C^* = \sum_{i=1}^{n_e} C_i^* \quad \text{and} \quad B^* = \sum_{i=1}^{n_e} B_i^*, \]  

where: \( n_e \) - number of elements in the considered continuous system.

Finally coordinates of the locked nodes have to be eliminated, as well as related rows and columns from the global matrices. The final form of the dynamics equation is

\[ A^* \cdot \dot{q}^* + B^* \cdot \ddot{q}^* + C^* \cdot q^* = P^*. \]  

4 CONSTRAINT EQUATIONS

A rotational joint is used to joint the multibody sub-system with the elastic structure of the beam. The joint is modelled by two constraint equations. Within the elastic structure, its horizontal position is constant for the revolute joint. To express it, multibody position of the connecting joint is considered as constant and it coincides with a position of a node \#j of the model of the elastic structure. Vertical position of the connection joint is modifiable, however,
vertical component of the node’s position coincides permanently with the vertical component of
the mechanism’s endpoint. Corresponding constraint equations are [11, 13, 17]
\[
h_i(q^b, q^t) = p_i(q^t) - q_i^t = 0 ; \quad h_2(q^b) = p_i(q^b) - L_n = 0 , \quad (23)
\]
where: \( \vec{p} \) – vector that express position of the endpoint at the multibody structure; \( p_i \) – \( i \)th component of the vector \( \vec{p} \); \( L_n \) – horizontal component of the position of the connection point measured along the elastic beam axis.

For velocity considerations, Jacobians \( J^b_v \) and \( J^b_h \) are introduced for vertical and horizontal components of the vector \( \vec{p} \) respectively (they are calculated in respect to multibody coordinates, only). Announced time derivatives of the constraint equations are [11, 13, 17]:
\[
\dot{h}_1 = J^b_v \cdot \dot{q}^b - \dot{q}_{2j-1}^t = 0 ; \quad \dot{h}_2 = J^b_h \cdot \dot{q}^b = 0 ; \quad (24)
\]
\[
\dot{h}_1 = J^b_v \cdot \dot{q}^b + A^b_v(q^b, \dot{q}^b) - \dot{q}_{2j-1}^t = 0 ; \quad \dot{h}_2 = J^b_h \cdot \dot{q}^b + A^b_h(q^b, \dot{q}^b) = 0 , \quad (25)
\]
where: \( J^b_v = \frac{\partial p_i}{\partial q^v_j} \); \( J^b_h = \frac{\partial p_i}{\partial q^h_j} \); \( A^b_v(q^b, \dot{q}^b) = \frac{\partial (J^b_v \cdot \dot{q}^b)}{\partial q^v_j} \cdot \dot{q}^v_j \); \( A^b_h(q^b, \dot{q}^b) = \frac{\partial (J^b_h \cdot \dot{q}^b)}{\partial q^h_j} \cdot \dot{q}^h_j \).

Dynamics eqs (9) and (22), are extended with Lagrange’s multipliers. It leads to [11, 13, 17]:
\[
M^b \cdot \ddot{q}^b + F^b + Q^b + J^{vT} \cdot \lambda = 0 ; \quad (26)
\]
\[
A^c \cdot \ddot{q}^c + B^c \cdot \dot{q}^c + C^c \cdot q^c + J^{cT} \cdot \lambda = P^c , \quad (27)
\]
where:
\[
J^v = \begin{bmatrix} J^v_1 \\ J^v_2 \end{bmatrix} ; \quad A^b = \begin{bmatrix} A^b_v \\ A^b_h \end{bmatrix} ; \quad J^v = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} .
\]

Next, introduced Lagrange’s multipliers, as well as the dependent coordinates, are eliminated. A modified version of the classical coordinate partitioning method [15] is proposed for it. Two different situations are considered

4.1 Dependent coordinates limited to a multibody structure, only
As the considered multibody sub-system has two degrees of freedom, all the multibody joint displacements are considered as dependent coordinates (i.e., all the multibody coordinates have to be eliminated). Equation (26) is used to evaluate the Lagrange multipliers
\[
\lambda = -J^{vT} \cdot M^b \cdot \ddot{q}^b - J^{vT} \cdot F^b - J^{vT} \cdot Q^b . \quad (28)
\]

Obtained multipliers are introduced in equation (27). It leads to
\[
A^c \cdot \ddot{q}^c + B^c \cdot \dot{q}^c + C^c \cdot q^c - J^{cT} \cdot J^{vT} \cdot M^b \cdot \ddot{q}^b - J^{cT} \cdot J^{vT} \cdot F^b - J^{cT} \cdot J^{vT} \cdot Q^b = P^c . \quad (29)
\]

Concerning the kinematics, with used of terms introduced in (26)-(27), equations (25) can be expressed in a common matrix form
\[
J^b \cdot \ddot{q}^b + A^b + J^c \cdot \ddot{q}^c = 0 , \quad (30)
\]
and accelerations of the dependent coordinates can be calculated
\[
\ddot{q}^b = -J^{vT} \cdot A^b \cdot \dot{J}^{vT} \cdot J^{vT} \cdot \dot{q}^c . \quad (31)
\]

Obtained accelerations are introduced in equation (29). It leads to
\[
A^c \cdot \ddot{q}^c + B^c \cdot \dot{q}^c + C^c \cdot q^c + \ddot{J}^{cT} \cdot J^{bT} \cdot M^b \cdot \ddot{q}^b + \ddot{J}^{cT} \cdot J^{bT} \cdot F^b + \ddot{J}^{cT} \cdot J^{bT} \cdot Q^b = P^c . \quad (32)
\]

526
After some rearrangement, it can be written as

\[ A^R \ddot{q}^c + B^R \cdot q^c + C^R \cdot \dot{q} = 0, \quad (33) \]

where:

\[ A^R = A^c + J^c \cdot J^c \cdot M^b \cdot J^b \cdot J^c \quad ; \quad B^R = B^c \quad ; \quad C^R = C^c \quad ; \]
\[ P^R = P^c - J^c \cdot J^c \cdot M^b \cdot J^b \cdot J^c \cdot \left( P^b + Q^b \right). \quad (34) \]

4.2 One of the multibody coordinates considered as independent

As an alternative, in the competitive model of the multibody joint displacements is considered as independent. One of the beam coordinates is considered as dependent, now. Equations (26)-(27) are expressed in a common matrix form

\[ M \cdot \ddot{q} + F + Q + J^T \cdot \dot{\lambda} = 0; \quad (35) \]

where:

\[ q = \begin{bmatrix} q^c \\ q^b \end{bmatrix} \quad ; \quad M = \begin{bmatrix} M^c & 0 \\ 0 & A^c \end{bmatrix} \quad ; \quad F = \begin{bmatrix} F^c \\ -P^c \end{bmatrix} \quad ; \quad Q = \begin{bmatrix} Q^c \\ 0 \end{bmatrix} \quad ; \quad J = \begin{bmatrix} J^c & J^b \end{bmatrix}. \quad (36) \]

Next, the classical coordinate partitioning method [15] is used to eliminate the dependant coordinates and the Lagrange’s coordinates, as well.

5 CONSIDERED SYSTEM

To verify effectiveness of the proposed damping, a set of numerical tests is performed. The presently considered system is similar to the one presented in [11, 12, 13, 14, 17]. Proposed multibody system is planar (Fig. 4a). It is composed of two bodies (arms). Its joints are rotational. The first arm is fixed to the reference body. Its fixing point is located 40 cm above of the forth node of the elastic structure. All the arms are made of light pipes. Length of the first is varying from 15 to 30 cm. The length of the second arm equals 22.4 cm. Similar inertial properties are proposed for the arms. Their masses equal 0.1 kg. The moments of inertia equal 0.000333 kg·m². The mass centres are shift 10 cm from the fixing points. The inertia products equal zero. According to the used length, the system configuration is close to its singular position (the second arm is almost aligned with the initial). In the neighbourhood of the singular position, the determinant of the Jacobian matrix is low. According to it small vertical component of the velocity (or acceleration) of the mechanism end point can effect in significant values of the joint velocities (accelerations). To obtain a symbolic form of kinematics and dynamics equations of the multibody structure, ROBOTRAN [16] program is used.

End of the mechanism’s second arm is joined rotationally to the elastic beam modelled as composed of finite elements. The length of the modelled beam equals 1m. A set of 50 finite elements is used to model the beam (Fig. 4b). Surface of its cross sections equals 0.001 m², its geometrical moment of inertia equals 10^{-8} m⁴ and density of its material is 7.8 \cdot 10^3 kg/m³. According to the illustrative reasons, relatively low Young modulus is supposed. For the considered tests, it equals 2.1 \cdot 10^9 Pa, only. The end of the multibody arm is joined to the fifteen
node of the beam (Fig. 4b). Model of dynamics of the beam is prepared according to equations presented in fifth section of the paper. The constraint equations are introduced as section 4.

5.1 Dependent coordinates limited to a multibody structure, only

Initially, all the multibody joint displacements are considered as dependent coordinates (i.e., all the multibody coordinates are eliminated). Dynamics of the system is described by (33). Initially, influence of the beam structural damping is tested. Equation (20) is used to determinate the damping matrix. Parameter \( a \) of the inertia influence equals zero. System behaviours are tested for different parameter \( \beta \) of the elasticity influence (Fig. 5a). To bypass influence of the disparity phenomenon, relatively long first arms are considered \((l_1=0.281 \text{ m})\). As it can be seen in the Fig. 5a, to obtain some relatively good damping, the influence parameter should be higher then \( 0.4 \% \). Next, influence of the disparity phenomenon is tested. Shorter first arm is considered \((l_1=0.188 \text{ m})\). Relatively low factor of the structural damping is considered \((\beta =0.001)\). Displacements of the fixing point (the one that connects the mechanism and the beam) are visualised in Fig. 5b, for activated and deactivated effects of the disparity. Displacements of some alternative point of the beam (node \#32) are visualised in Fig. 5c.

Figure 5: Vertical displacement of a selected node of the beam: correlation between the structural damping and the displacement of the fixing point that connects the mechanism and the beam \((l_1=0.281 \text{ m})\) (a); correlation between the length of the first arm and the displacement of the fixing point that connects the mechanism and the beam \((\beta =0.001)\) (b); correlation between the length of the first arm and the displacement of a selected node of the beam \((\beta =0.001)\) (c)

Some interesting indications of the system behaviour can be observed when mass associated to one of the beam nodes (the node connected to the endpoint of the mechanism) is observed after the elimination of the multipliers and the dependent coordinates. Its mass is presented in Fig. 6a. It indicates that the mass is changing its value significantly depending on the obtained position of the endpoint of the mechanism. When the rotation angle at the second joint of the mechanism approaches to its zero value, the mass is growing significantly to some “unrealistically” high values. The maximal obtained value in the test equals 13067 kg. The narrow picks of the node inertia are observed for some extremely short periods of time, when the joint angle obtains radically small values, only. The rapid growth of the inertia can effect in a short time modification of forms of the system vibrations. Higher frequency modes are excited and well damped. Total energy of the system diminishes significantly.

Figure 6: Behaviour of the system with shorter length of the first arm \((l_1=0.188 \text{ m}, \beta =0.001)\): mass associated to the locked node (after the multipliers reduction) (a); displacement of the fixing point that connects the mechanism and the beam (b); rotation angle at the second joint of the multibody structure (c)
Obtained numerical model participates in numerical integrations badly. Especially ineffective is the procedure of solution of the nonlinear system of constraint equations for joint positions. When the numerical procedure of integration tests some of the potentially next coordinates, it can happen for the tested value that the node displacement is out of the range of the potential positions of the mechanism endpoint. Constraint equations may not be satisfied and the numerical procedure fault, stopping the calculations. To prevent against it, initial conditions have to be selected in a way that prevent the system to come too close to the singular position. Other numerical penalty actions are excluded from considerations. They can result in non natural behaviours from a point of view of the system energy, especially when combined with the strong nonlinear effects observed at the neighbourhood of the singularity.

5.2 One of the multibody coordinates considered as independent

To prevent again the numerical problems announced in the previous sub-section, an alternative selection of the dependent coordinates is proposed. At the present test, the first joint of the mechanism sub-system is considered as independent. Instead of it, position of the node #15 (the node connected to the endpoint of the mechanism) is considered as the dependant coordinate. With the announced selection, solution of the nonlinear constraint equations can be performed correctly for any of the potential sets of the independent coordinates. The risk of the numerical break is eliminated.

According to the idea proposed in the previous sub-section, a comparison of the behaviour is made for long and short length of the first arm of the double-bar mechanism (Fig. 7, 8). The shorter arm corresponds to the case when the mechanism singular position is obtained within the range of the realised vibrations.

As it can be seen in the Fig. 7, 8, with the announced disparity effects, vibrations are significantly better damped, even in the cases of low structural damping in the beam. The reduced mass associated to the node #15, may not be observed, as it is not present directly in the list of the calculated quantities, now. However, the trans-pass over the singularity position is possible, now (a dotted line in sub-plots of Fig. 9).
Figure 9: Rotation angle at the second joint of the multibody structure: factor of the structural damping $\beta$ =0.01 (a); factor of the structural damping $\beta$ =0.001 (b); factor of the structural damping $\beta$ =0.00001 (c)

6 SUMMARY AND PERSPECTIVES

As it is presented in the paper, the use of the modal disparity effects looks as an effective damping method for low frequency, low damped vibrations present in a continuous system. The method can be employed especially in all the situations, when additional damping elements are precluded because of some of the technical reasons. The energy of unrequited vibrations could be distributed amount different (including these better damped) modes of vibrations and dissipated quicker.

To model the considered situation, a hybrid multibody/finite element model is necessary. To connect these structurally different sub-models, constraint equations are proposed, as well as modified version of the classical Lagrange multipliers elimination technique. The introduced methodology is found as an effective modelling method. Calculation times are relatively short. The model based on the constraint equations is an interesting an effective alternative to the classical elastic contacts.

Presented results indicate that some future investigations are necessity. As an example, the influence of the mechanism inertia has to be detailed. Alternative singular configurations as well as alternative mechanisms should be tested. The influences of the joint frictions and backslashes have to be introduced as well as optimal localization of the fixing point that connect the beam and the mechanism.

REFERENCES


[8] I. H. Wyllie, Discomfort from sinusoidal oscillation in the roll and lateral axes at frequencies between 0.2 and 1.6 hz, the 40th United Kingdom Conference on Human Response to Vibration, Liverpool, England, 2005, pp. 13-15


Dynamics of synchronization of rotational motion of contacting triple-body systems

Viktor Berbyuk

Department of Applied Mechanics
Chalmers University of Technology
Hörsalsvägen 7A, SE-412 96 Göteborg, Sweden
viktor.berbyuk@chalmers.se

ABSTRACT

Dynamics of synchronization of rotational motion of contacting triple-body system is studied. It is assumed that all bodies of the system can rotate completely independently of one another and perform rotational motion about the same axis under the action of external and internal loads. The direct dynamics synchronization problem for the contacting triple-body system is stated and the algorithm of the solution of the problem is presented. To determine the synchronization time the transcendental equation is derived. The torques between contacting interfaces are determined by solving two integral equations. Finally, by knowing synchronization time and the torques between contacting interfaces, the rotational motion of the system during synchronization process is determined by integration of the governing equations with prescribed initial conditions. It was shown that the solution of the direct dynamics problem is not unique and the synchronizing torques can be subject to optimization. The obtained results are applied for a generic synchronizer mechanism of manual transmission systems. Assuming that the drag torque, the vehicle resistance torque, the control torque applied to the selector sleeve, and the synchronizing torques at the contacting interfaces are substantially constant during synchronization the solution of the direct dynamics synchronization problem for a generic synchronizer is obtained in analytical form. Sensitivity analysis of the synchronization time and the gear shifting comfort with respect to the control torque applied to the selector sleeve and the driveline vibration has presented.

Keywords: Synchronizer mechanism, Transmission system, Synchronization time, Gear shifting comfort, Driveline vibration.

1 INTRODUCTION

Different engineering systems often comprise power transmission with gearbox. To perform gear shifting synchronizer mechanisms have been developed. Transmission gear shifting improvement with respect to smooth, quick and energy efficient synchronizer’s performance, is still one of the major concern area for automotive industry, see e.g. [1-6]. Typical synchronization process includes different phases of relative motion between functional components of synchronizer and gearbox that should be described by taking into account contact phenomenon in contacting interfaces, lubrication, temperature, and other issues, see e.g. [4-13]. So, the dynamics of synchronization processes is a challenge. Solution to this engineering problem requires deep insight into the dynamics and optimization of multibody system used to model the synchronization processes in power transmissions.

Aiming modelling and analysis of dynamics of synchronizing processes in vehicle transmission systems the paper is focused on main synchronization phase, namely the case when all bodies of the synchronizer mechanism are in contact. As an engineering model of a generic synchronizer the contacting triple-body system is proposed. It is assumed that all bodies of the system can rotate completely independently of one another and perform rotational motion about the same
axis under the action of external and internal loads. The direct dynamics synchronization problem for the contacting triple-body system (CTBS) is considered. For given external loads it is required to determine the torques between contacting interfaces and the rotational motion of the system that all together satisfy the equations of motion, prescribed initial conditions and guarantee synchronization of rotational speeds of all bodies for the final time.

2 MODEL AND PROBLEM STATEMENT

Consider the multibody system depicted in Figure 1. The system consists of three rigid bodies: B1, B2, and B3. It is assumed that all bodies of the CTBS can rotate completely independent of one another and perform rotational motion about the same axis Ox under the action of external and internal loads. The external load comprises the prescribed torque \( M_s(t) \) and the control torque \( M_f(t) \) both acting on the body B1, and the prescribed torque \( M_d(t) \) acting on the body B3. The internal load comprises the synchronizing torque \( M_s(t) \) acting at the contacting interface between the body B1 and the body B2, and the synchronizing torque \( M_f(t) \) acting at the contacting interface between the body B2 and the body B3.

With above mentioned assumptions the considered CTBS has three degrees of freedom. Using free body diagram the equations of rotational motion of the system in question can be written as follows (see Figure 1):

\[
J_v \ddot{\theta}_v(t) = M_s(t) - M_v(t) + M_f(t)
\]  

(1)

\[
J_r \ddot{\theta}_r(t) = M_r(t) - M_r(t)
\]  

(2)

\[
J_d \ddot{\theta}_d(t) = -M_d(t) - M_f(t)
\]  

(3)
Here $\theta_s, \theta_r, \theta_g$ are the angular coordinates of the bodies B1, B2 and B3, respectively. $J_s, J_r, J_d$ are the moments of inertia of the bodies B1, B2 and B3, respectively.

Let assume that at the initial time instant, $t = 0$, the phase state of the CTBS is prescribed by the following conditions:

\[
\begin{align*}
\theta_s(0) &= \theta_{s0}, \quad \dot{\theta}_s(0) = \omega \\
\theta_r(0) &= \theta_{r0}, \quad \dot{\theta}_r(0) = r_i \omega \\
\theta_g(0) &= \theta_{g0}, \quad \dot{\theta}_g(0) = r \omega
\end{align*}
\]

Here in the expressions (4) - (6), $\theta_{s0}, \theta_{r0}, \theta_{g0}, \omega, r_i, r$ are given numbers, and $r_i > 0, r > r_i, r > 1$.

The following direct dynamics synchronization problem (DDSP) can be stated.

**Problem DDSP:** Let we are given by the external prescribed torque $M_v(t)$ acting on the body B1

\[
M_v = M_v(t), \quad t \geq 0
\]

the external prescribed torque $M_d(t)$ acting on the body B3

\[
M_d = M_d(t), \quad t \geq 0
\]

the control torque applied to the body B1

\[
M_f = M_f(t), \quad t \geq 0
\]

and the initial conditions (4)-(6).

It is required to determine synchronizing torques between the contacting interfaces, $M_s(t), M_r(t), M_g(t)$, the rotational motion of the CTBS, $\theta_s(t), \theta_r(t), \theta_g(t)$, and the time instant, $t = t_s$, that all together satisfy the equations (1)-(3), the initial conditions (4)-(6), and the following final conditions

\[
\dot{\theta}_s(t_s) = \dot{\theta}_r(t_s) = \dot{\theta}_g(t_s) = r \omega
\]

Note that the angular coordinates at the final time instant $t = t_s$, i.e. the values $\theta_s(t_s), \theta_r(t_s), \theta_g(t_s)$ are free. The parameter $t_s$ will be termed the synchronization time.
3 ALGORITHM OF THE SOLUTION OF THE DDSP

Assuming that the torques $M_v, M_d, M_F, M_s, M_r$ are the functions of time the following notations can be introduced:

\[ M_{v1}(t) = \int_0^t M_v(\tau) d\tau, \quad M_{v2}(t) = \int_0^t M_v(\tau) d\tau \]
\[ M_{d1}(t) = \int_0^t M_d(\tau) d\tau, \quad M_{d2}(t) = \int_0^t M_d(\tau) d\tau \]
\[ M_{f1}(t) = \int_0^t M_F(\tau) d\tau, \quad M_{f2}(t) = \int_0^t M_F(\tau) d\tau \]  \hfill (11)

By integrating the equations (1)-(3) with initial conditions (4)-(6) and using the notations (11), the angular velocities and angular coordinates of the rotating bodies of the CTBS are expressed as follows:

\[ \dot{\theta}_v(t) = \frac{[M_{v1}(t) - M_{v1}(t) + M_{f1}(t)]}{J_v + \omega} \]
\[ \dot{\theta}_d(t) = \frac{[M_{d1}(t) - M_{d1}(t)]}{J_d + r_\omega} \]  \hfill (12)
\[ \dot{\theta}_g(t) = \frac{[-M_{d1}(t) - M_{d1}(t)]}{J_d + r_\omega} \]

Using the conditions of synchronization (10), the expressions (12) and the notations (11), the following system of the integral-transcendental equations for unknown synchronization time $t_s$ and synchronizing torques $M_v(t), M_d(t)$ are obtained:

\[ \int_0^{t_s} [M_v(t) - M_{v2}(t) + M_{f2}(t)] dt = J_v \omega (r - 1) \]  \hfill (14)
\[
\int_{0}^{t} [M_s(t) - M_s(t)] dt = J_s \omega (r - r_s) 
\]  \hspace{1cm} (15)

\[
\int_{0}^{t} [-M_{ds}(t) - M_s(t)] dt = 0 
\]  \hspace{1cm} (16)

By summing the equations (14)-(16) we will get

\[
\int_{0}^{t} [M_{rs}(t) - M_{gs}(t) - M_{ds}(t)] dt = \alpha \omega J_s (r - 1) + J_s (r - r_s) 
\]  \hspace{1cm} (17)

Analysis of the obtained equation (17) shows that for the considered CTBS and the synchronization conditions the synchronization time \( t_s \) is determined only by the external loads and the inertial properties of the body B1 and the body B2, and does not depend on the internal synchronizing torques. It means that within the considered assumptions the quickness of the synchronization of the CTBS does not depend on the inertial properties of the body B3 as well as the internal load distributions at the contacting interfaces between the body B1 and the body B2, and between the body B2 and the body B3.

It is also obvious that the external torques, \( M_{ds}(t), M_{rs}(t), M_{gs}(t) \), can’t be chosen arbitrary. These torques must be chosen in that way that it is guaranteed the existing at least one positive root of the transcendental equation (17) with respect to parameter \( t_s \). In this case by knowing the synchronization time \( t_s \), the corresponding internal synchronizing torques \( M_s(t), M_g(t) \) are determined by solving the equations (15), (16).

As it follows from analysis of the equations (15) and (16) the functions \( M_s(t), M_g(t) \) are not determined uniquely. That is, if for the prescribed external loads (7)-(9) the solution to the Problem DDSP exist, then the solution is not unique. Consequently, the solution of the direct dynamics synchronization problem for the considered CTBS can be subject to optimization.

Finally, knowing the synchronization time \( t_s \) and the toques at the contacting interfaces \( M_s(t), M_g(t) \), the rotational motion of the CTBS that satisfy the boundary conditions (4)-(6), (10) is obtained from expressions (12) and (13). It concludes the algorithm of the solution of the formulated direct dynamics synchronization problem for the CTBS.

4 DYNAMICS OF A GENERIC SYNCHRONIZER

Let assume that the considered CTBS models a generic synchronizer mechanism during the main synchronization phase in the vehicle manual transmission. The bodies B1, B2 and B3 will be termed the selector sleeve, the blocker ring and the gearwheel, respectively. The external load comprises: the torque \( M_{rs}(t) \), the vehicle resistance acting on the output side of the gearbox; the control torque \( M_{rs}(t) \) acting on the selector sleeve, (the torque arising due to the external axial force applied to selector level); and the torque \( M_{ds}(t) \), the drag torque acting on the input side of the gearbox. The internal load comprises the synchronizing torque \( M_s(t) \) acting at the contacting interface between the selector sleeve and the blocker ring and the synchronizing torque \( M_g(t) \) acting at the contacting interface between the blocker ring and the gearwheel.
Now in (4)-(17) $\omega$ is the initial angular velocity of the output side of the synchronizer; $r$ is the gear step, i.e. the ratio of the initial angular velocity of the input and the output sides of the synchronizer; $J_d, J_s$ are the equivalent moment of inertia of the input (engine clutch) and output (all vehicle components including that of vehicle itself) sides of the gearbox, respectively; $J_r$ is the moment of inertia of the blocker ring of the synchronizer mechanism.

Following [11, 12] and assuming that the resistance torque on the synchronizer owing to the vehicle inertia, the drag torque and the synchronizing torques at the contacting interfaces are substantially constant during the synchronization process consider the Problem DDSP with the following assumptions:

$$M_{vq}(t) = M_{v0} = \text{constant}, \quad M_{de}(t) = M_{d0} = \text{constant}, \quad t \geq 0$$

$$M_s(t) = M_{s0} = \text{constant}, \quad M_s(t) = M_{s0} = \text{constant}, \quad t \geq 0$$

(18)

Let assumed also that the external control torque $M_F(t)$ has a constant value during synchronization, i.e.

$$M_F(t) = M_{F0} = \text{constant}, \quad t \geq 0$$

(19)

With assumptions (18) and (19) by using the expressions (13) with notations (11) and the equations (15)-(17), the solution to the direct dynamics synchronization problem for the generic synchronizer is determined as follows:

$$t_s(M_{F0}) = \frac{\omega J_s(r-1) + J_r(r-r_s)}{M_{F0} - M_{s0} - M_{d0}}$$

(20)

$$M_{v0} = -M_{d0}$$

(21)

$$M_{s0} = \frac{J_s(1-r)M_{d0} + J_r(r_s-r)(M_{F0} - M_{s0})}{J_s(r-1) + J_r(r-r_s)}$$

$$\theta_s(t) = t^2(M_{s0} - M_{s0} + M_{F0}) / (2J_s) + \omega t + \theta_{s0}$$

(22)

$$\theta_r(t) = t^2(M_{r0} - M_{r0}) / (2J_r) + r_s \omega t + \theta_{r0}$$

$$\theta_g(t) = r \omega t + \theta_{g0}$$

By taking into account that $r > 1, r > r_s$ and the synchronization time must be positive, from (20) follows the restriction on the external control torque applied to the selector sleeve.
\[ M_{v_0} + M_{d_0} < M_{F_0} \leq M_{F}^{\text{max}} \]  \hspace{1cm} (23)

Here in (23) \( M_{F}^{\text{max}} \) is the maximal admissible value of the external control torque \( M_{F}(t) \).

From the expression (20) it follows that the larger torque \( M_{F_0} \) applied to the sleeve the smaller synchronization time \( t_s \) will be. In case of restriction (23) the minimal value of the synchronization time, \( t_{s_{\text{min}}} \), is determined by the expression

\[ t_{s_{\text{min}}} = \frac{\alpha [J_r (r-1) + J_c (r-r_c)]}{M_{F_{\text{max}}} - M_{v_0} - M_{d_0}} \]  \hspace{1cm} (24)

### 4.1 Driveline vibration and synchronization time

Consider the dynamics of a generic synchronizer mechanism for the case when the vehicle resistance acting on the output side of the gearbox, \( M_{v_0} \), is subject to periodic excitations.

Namely, let us assume that the torque \( M_{v_0}(t) \) is prescribed as follows

\[ M_{v_0}(t) = M_{v_0} + A_v \sin(\omega_v t + \varphi_v) \]  \hspace{1cm} (25)

Here \( A_v \), \( \omega_v \), \( \varphi_v \) are the amplitude, the frequency and the phase angle of the periodic excitations applied to the output side of the gearbox (driveline vibrations).

The drag torque, the synchronizing torques at the contacting interfaces as well as the external control torque applied to the selector sleeve are assumed to be substantially constant during the synchronization and satisfy the conditions (18) and (19).

With the above assumptions and the expression (25) the equation (17) can be written as follows

\[ t_s \left[ [M_{F_0} - M_{v_0} - M_{d_0} - A_v \sin(\omega_v t + \varphi_v)] dt = \alpha [J_r (r-1) + J_c (r-r_c)] \right] \]  \hspace{1cm} (26)

By introducing the notation

\[ t_{s_0} = \frac{\alpha [J_r (r-1) + J_c (r-r_c)]}{M_{F_0} - M_{v_0} - M_{d_0}} \]  \hspace{1cm} (27)

the equation (26) gives the following relation

\[ t_s = t_{s_0} + A_v \frac{\cos(\varphi_v) - \cos(\omega_v t_s + \varphi_v)}{\omega_v (M_{F_0} - M_{v_0} - M_{d_0})} \]  \hspace{1cm} (28)
Here \( t_s \) and \( t_{s0} \) are the synchronization times with and without driveline vibrations, respectively, (see and compare the expressions (20) and (27)).

Let us assume that we are given by the periodic excitation (25) with zero phase angle, i.e. \( \varphi_v = 0 \). Then as it follows from the expression (28) we have the relation

\[
t_s = t_{s0} + \frac{A_v [1 - \cos(\omega t_s)]}{\omega (M_{F0} - M_{v0} - M_{d0})}
\]  

(29)

Analysis of the relation (29) shows that for all feasible synchronization processes with the admissible control torque \( M_{F0} > M_{v0} + M_{d0} \) the driveline vibration prescribed by the expression \( M_{v} (t) = M_{v0} + A_v \sin \omega t \) will increase the synchronization time, i.e. \( t_s > t_{s0} \), for any values of the amplitude and the frequency of the periodic excitation.

Consider the periodic excitation (25) with the phase angle \( \varphi_v = -\pi / 2 \), i.e. the case when the driveline vibration is prescribed by the expression

\[
M_{v} (t) = M_{v0} - A_v \cos \omega t
\]  

(30)

From the expression (28) we can obtain

\[
t_s = t_{s0} + \frac{A_v \sin \omega t_s}{\omega (M_{F0} - M_{v0} - M_{d0})}
\]  

(31)

Analysis of the relation (31) shows that one can expect that the driveline vibration prescribed by the expression (30) can shorten the synchronization time \( t_s \). For instance, if \( \omega \ll 1 \), then \( \sin \omega t_s = \omega t_s \) and the relation (31) gives the following

\[
t_s = t_{s0} - \frac{A_v}{M_{F0} - M_{v0} - M_{d0}}
\]  

(32)

i.e. the inequality \( t_s < t_{s0} \) is valid for all \( A_v < 0 \) and \( M_{F0} > M_{v0} + M_{d0} \).

4.2 Time-comfort optimal gear shifting

Quick gear shifting can lead to clashing, double bump, and other phenomena in a transmission system which effect negatively gear shifting comfort.

Let us assume that the total inertia load acting on the synchronizer mechanism

\[
Q_{\text{e}}(M_{F0}) = J_e \dot{\theta}_e (M_{F0}) + J_s \dot{\theta}_s (M_{F0}) + J_d \dot{\theta}_d (M_{F0})
\]  

(33)

can be used as an indicator of the gear shifting comfort. Moreover, it is assumed that the lowest value of the function (33) corresponds to the best comfort.
With the expressions (22) the function (33) becomes

$$Q_c(M_{F_0}) = M_{F_0} - M_{v_0} - M_{d_0}$$  \hspace{2cm} (34)$$

Analysis of the functions $t_c(M_{F_0})$ and $Q_c(M_{F_0})$ expressed by (20) and (34), shows that there exist the control torque $M'_{F_0}$ such that the following system of variational equations are obeyed

$$\min_{M_{F_0} \in U_F} t_c(M_{F_0}) = t_c(M'_{F_0})$$

$$\min_{M_{F_0} \in U_F} [Q_c(M_{F_0})] = Q_c(M'_{F_0})$$  \hspace{2cm} (35)$$

Here in (35) $U_F$ is a set of admissible control torques defined by the inequalities (23).

The variational equations (35) together with the inequalities (23) constitute the time-comfort Pareto optimal control problem for considered generic synchronizer mechanism. The solution to this problem is given by the following expressions [14]:

$$M'_{F_0} = M_{v_0} + M_{d_0} \mp \sqrt{M_{F_0} \max (M_{F_0} \max - M_{v_0} - M_{d_0})}$$

$$t_c(M'_{F_0}) = \frac{\alpha [J_s (r-1) + J_f (r-r_f)]}{\sqrt{M_{F_0} \max (M_{F_0} \max - M_{v_0} - M_{d_0})}}$$

$$Q_c(M'_{F_0}) = \sqrt{M_{F_0} \max (M_{F_0} \max - M_{v_0} - M_{d_0})}$$  \hspace{2cm} (36)$$

5 CONCLUSIONS

In the paper a contacting triple-body system dynamics is studied with focus on synchronization of its rotational motion. The direct dynamics synchronization problem is stated and the algorithm of the solution of the problem has developed. It was found that within the considered assumptions the synchronization time is determined only by the external loads and does not depend on internal synchronizing torques at contacting interfaces. It is also proved that for given external loads the solution of the direct dynamics synchronization problem is not unique and can be subject to optimization.

The obtained results have been applied to study the dynamics of a generic synchronizer mechanism of a vehicle transmission system. The solution of the direct dynamics synchronization problem was obtained in an analytical form. Sensitivity of the synchronization time with respect to driveline periodic excitations was studied. It was found that the values of the phase angle, the amplitude and the frequency of driveline periodic excitations significantly affect the synchronization time. Driveline vibrations can increase or decrease the synchronization time. It was shown that the synchronization time is heavily sensitive with respect to the phase angle of the driveline periodic excitations.

For the considered generic synchronizer mechanism the problem of time-comfort optimal control of gear shifting was stated and its solution has presented.
ACKNOWLEDGMENTS

This paper was written in the context of the project “Robust and optimized synchronization of transmissions” funded by the Sweden’s Innovation Agency, VINNOVA, Scania CV AB and Volvo GTT, (Project reference number: 2012-04619).

REFERENCES

Forward dynamics of selected spatial one-dof linkage mechanisms with the Dahl friction model in revolute joints

Andrzej Harlecki, Andrzej Urbaś
Faculty of Mechanical Engineering and Computer Science
University of Bielsko-Biała
Willowa 2, 43-309 Bielsko-Biała, Poland
[aharlecki, aurbas]@ath.eu

ABSTRACT
Dynamic analysis of a one-dof RSRRR spatial linkage mechanism, including four revolute joints R and one spherical joint S, is presented in the article. It is assumed that friction occurs in the revolute joints R, whereas the spherical joint S was treated as an ideal joint. The mechanism in the form of a closed-loop kinematic chain was divided by the cut-joint technique into two open-loop kinematic chains in the place of the spherical joint S. Joint coordinates and homogeneous transformation matrices were used to describe the geometry of the chains obtained. Equations of the chains’ motion were derived using the formalism of Lagrange equations. In order to determine unknown values of the reaction forces acting in the cutting place, additional algebraic equations of geometrical constraints were introduced to complete the equations of motion. As a consequence, a set of differential-algebraic equations was obtained. In order to solve these equations, a procedure was applied based on double differentiation of the constraints’ equations formulated in relation to time. The Dahl friction model was assumed in order to take into account friction in the revolute joints. To determine the values of friction torques in the revolute joints, in each integrating step of the equations of motion the reaction forces and torques acting in these joints were calculated using the recursive Newton-Euler algorithm taken from robotics. A special model of the revolute joints was developed for the requirements of the method. Some examples of results of the numerical calculations made here are presented in the conclusions of the article.

Keywords: Forward dynamics, Linkage mechanism, Lagrange equations, Recursive Newton-Euler algorithm, Dahl friction model.

LIST OF IMPORTANT SYMBOLS

- $c$: kinematic chain index
- $g$: acceleration of gravity
- $(c, p)$: symbol of link (joint) p in chain $c$
- $l^{(c,p)}$: length of link $(c, p)$
- $m^{(c,p)}$: mass of link $(c, p)$
- $n_b^{(c)}$: number of links in chain $c$
- $\mathbf{r}_A^{(c,p)}$: vector of position of point $A$ in the local coordinate system
- $\mathbf{H}^{(c,p)}$: inertia matrix $4 \times 4$ of link $(c, p)$
- $\mathbf{T}^{(c,p)}$: transformation matrix $4 \times 4$ from the local coordinate system of link $(c, p)$ to the system of link $(c, p-1)$
- $\mathbf{T}_i^{(c,p)}$: transformation matrix $4 \times 4$ from the local coordinate system of link $(c, p)$ to the global reference system $\{1,0\}$

$$\mathbf{T}_i^{(c,p)} = \frac{\partial \mathbf{T}^{(c,p)}}{\partial q_i^{(c,p)}}$$
\[
T_{ij}^{(c,p)} = \frac{\partial^2 T^{(c,p)}}{\partial q_i^{(c,p)} \partial q_j^{(c,p)}}
\]

dof – degree(s) of freedom

1 INTRODUCTION

The subject of analysis presented in this paper is a selected class of one-dof spatial linkage mechanisms. These mechanisms contain neither a redundant dof nor a passive constraint. In their structure, one spherical joint S (with three-dof) and a few revolute joints R (sometimes also one prismatic joint P) with one-dof occur. Some mechanisms contain a universal joint U, being the system of two revolute joints R. The dynamics of this class of mechanisms was analysed, e.g. in paper [1, 2, 3]. In the literature one can find relatively few works devoted to the analysis of dynamics of spatial linkage mechanisms with friction in joints. One can include here, besides the already cited work [3], also, e.g. publications [4, 5]. The authors of paper [3] took into account the bending and torsional flexibility of links as well as backlash. Friction in joints was analysed using the LuGre friction model [6]. The author of monograph [4] analysed the dynamics of a one-dof RRUUR spatial linkage mechanism with friction in selected joints using the NEWEUL computing program, which was developed and is applied mainly in Germany. The authors of work [5] analysed the dynamics of the steering system of personal cars, with the McPherson suspension being the RSSPSSR spatial linkage mechanism with three-dof (two of them are redundant ones). They took into account stick-slip motion in the prismatic joint P, i.e. both the phases of static and kinetic friction were analysed here. The authors of work [7] presented a method of dynamic analysis of parallel manipulators, in the form of closed-loop kinematic chains, by taking into account the phenomenon of stick-slip motion in their joints. As an example they analysed a 3-dof 3RPS parallel manipulator which consisted of a fixed base and a moving platform that was guided by free linear actuators. Many more papers can be found in the literature which are devoted to the dynamic analysis of planar linkage mechanisms with friction in joints. One of the first papers in this field was the often quoted publication [8], however, one of the most recent papers is publication [9], whose authors used the Dahl friction model.

2 DESCRIPTION OF THE DYNAMIC ANALYSIS METHOD

In the method presented here, friction is taken into account in all of the revolute joints R and in the prismatic joint P (if it occurs in the structure of the mechanism), whereas the spherical joint S is treated as an ideal joint. It is assumed that the mechanism is cut in the place of the spherical joint S, thus obtaining two open kinematic chains joined with the fixed base. In the cutting place applicable reaction forces, acting on the formed chains, are introduced. The Denavit-Hartenberg notation [10] is used to describe the geometry of the chains. This notation is based on homogeneous transformation matrices 4 x 4 and joint coordinates (defining the relative position of the links) according to the methodology presented in work [11], which deals with a description of the geometry of the robots’ manipulators. It can be seen quite clearly that the structure of the chains obtained is analogical to the structure of the manipulators, which are modelled in the form of open-loop kinematic chains. Thus the procedure regarding formulating equations of the chains’ motion can be analogical to the approach which is used in the case of the dynamic analysis of the manipulators. In the method accepted here, the equations of motion of chains are determined by using the formalism of Lagrange equations on the basis of the algorithms provided in monograph [12]. Additional equations of geometrical constraints are formulated in order to determine unknown reaction forces at cut-joint S. These equations (in the form of algebraic equations) are joined with the equations of motion (in the form of ordinary differential equations). A system of differential-algebraic equations is formed in such a way. To determine the values of reaction forces and torques in each integrating step of the equations of motion and then the values of friction torques acting in the joints with friction, additional calculations were performed using the recursive Newton-Euler algorithm as presented in monograph [11]. In practice, solving the differential-algebraic equations is relatively complex.
and usually by double differentiation of the algebraic equations of the constraints in relation to time they are transformed to a system of ordinary differential equations. Such a way was also assumed in the case of the method proposed here. Equations obtained in such a way can be solved by using any numerical methods with a relatively large integration step. If it turns out to be necessary, a method of constraint stabilisation can be used to improve the accuracy of the results of the numerical calculations [13].

3 EXAMPLE OF THE ANALYSIS

The method described here was used to analyse (taken from work [1]) a spatial RSRRR linkage mechanism built of four movable links joined with an immobile base. The mechanism contains neither redundant dof nor passive constraints. The number of dof of the mechanism is: \(4 \cdot 6 - 4 \cdot 5 - 1 \cdot 3 = 1\). The driving link of the mechanism is loaded by the driving torque \(T_{\text{dr}}^{(1,1)}\) and reduced resistance torque \(T_{\text{rev}}^{(1,1)}\) (Fig. 1).

The mechanism considered here was divided in the place of the spherical joint S, and this resulted in obtaining two open-loop kinematic chains joined with the immobile base: 1 – formed by link \((1,1)\), and 2 – formed by links \((2,1)\), \((2,2)\) and \((2,3)\). The links in the kinematic chains are interconnected by revolute joints R: \((1,1)\) – in chain 1, and \((2,1)\), \((2,2)\), \((2,3)\) – in chain 2. As is shown in Fig. 2, the local coordinate systems, formed by right-hand-oriented versors, were attached according to the guidelines of the Denavit-Hartenberg notation to particular links (including the immobile base). The fixed coordinate system \((1,0)\), related to chain 1, was understood as the global reference system, and system \((2,0)\), related to chain 2, was the auxiliary reference system.

The motion of the chains is described by the vectors of joint coordinates \(\mathbf{q} = \left[ q_{j} \right]_{c=1}^{n_c}, \mathbf{r} = \left[ r_{j} \right]_{c=1}^{n_r} \), so:

- for chain 1: \(\mathbf{q}^{(1,1)} = \left[ q_{j}^{(1,1)} \right]_{c=1}^{n_c}, \mathbf{r}^{(2,1)} = \left[ r_{j}^{(2,1)} \right]_{c=1}^{n_r}\)

- for chain 2: \(\mathbf{q}^{(2,3)} = \left[ q_{j}^{(2,3)} \right]_{c=1}^{n_c}, \mathbf{r}^{(2,2)} = \left[ r_{j}^{(2,2)} \right]_{c=1}^{n_r}\).

The homogeneous transformation matrices from the local systems attached to the links to the global reference system \(\{1,0\}\) were determined according to the relationship:

\[
\mathbf{T}^{(c,p)} = \mathbf{T}^{(c,p-1)} \mathbf{T}^{(c,p)}
\]
where: \( \mathbf{T}^{(c,0)} = \mathbf{1} \),

\[
\mathbf{T}^{(1,1)} = \begin{bmatrix}
c_{\psi}^{(1,1)} & -s_{\psi}^{(1,1)} & 0 & 0 \\
s_{\psi}^{(1,1)} & c_{\psi}^{(1,1)} & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix},
\mathbf{T}^{(2,1)} = \begin{bmatrix}
s_{\psi}^{(2,1)} & c_{\psi}^{(2,1)} & 0 & 0 \\
c_{\psi}^{(2,1)} & -s_{\psi}^{(2,1)} & 0 & f^{(2,1)} \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix},
\mathbf{T}^{(2,2)} = \begin{bmatrix}
c_{\psi}^{(2,2)} & -s_{\psi}^{(2,2)} & 0 & 0 \\
s_{\psi}^{(2,2)} & -c_{\psi}^{(2,2)} & 0 & f^{(2,2)} \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix},
\mathbf{T}^{(2,3)} = \begin{bmatrix}
s_{\psi}^{(2,3)} & c_{\psi}^{(2,3)} & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix},
\]

\( s_{\psi}^{(a,\beta)} = \sin(\psi^{(a,\beta)}), \quad c_{\psi}^{(a,\beta)} = \cos(\psi^{(a,\beta)}) \).

The reaction forces \( f_{S,x}, f_{S,y}, f_{S,z} \) and \(-f_{S,x}, -f_{S,y}, -f_{S,z}\), acting on chain 1 and 2, respectively, in accordance with the versor directions of the global reference system \( \{1,0\} \), were applied at cut-joint S, and precisely in centre S of this joint.

4 SYNTHESIS OF EQUATIONS OF MOTION AND THE ALGORITHM TO SOLVE THEM

The equations of motion of both open-loop kinematic chains can be presented in the following form:

\[
\begin{bmatrix}
\mathbf{A}^{(1,1)} & 0 & -\mathbf{D}^{(1,1)} \\
0 & \mathbf{A}^{(2,3)} & \mathbf{D}^{(2,3)} \\
\mathbf{D}^{(1,1)^T} & -\mathbf{D}^{(2,3)^T} & 0
\end{bmatrix}
\begin{bmatrix}
\ddot{\mathbf{q}}^{(1,1)} \\
\ddot{\mathbf{q}}^{(2,3)} \\
\mathbf{f}_S
\end{bmatrix}
= \begin{bmatrix}
\mathbf{e}^{(1,1)} + \mathbf{t}^{(1,1)}_{dr} - \mathbf{s}^{(1,1)}_{dr} - \mathbf{s}^{(1,1)}_{f} \\
\mathbf{e}^{(2,3)} - \mathbf{s}^{(2,3)}_{f} \\
\mathbf{e}^{(1,2)}
\end{bmatrix},
\tag{2}
\]

where:

\( \mathbf{A}^{(c,n^{(c)})}_{ij} = \{ \mathbf{A}^{(c,n^{(c)})}_{ij} \}_{i,j=1,\ldots,n^{(c)}}, \)

\( \mathbf{A}^{(c,n^{(c)})}_{ij} = \sum_{p=\max[i,j]}^{n^{(c)}} \mathbf{T}_i^{(c,p)} \mathbf{H}^{(c,p)} \mathbf{T}_j^{(c,p)^T} \),

\( \mathbf{e}^{(c,n^{(c)})}_{k} = \{ \mathbf{e}^{(c,n^{(c)})}_{k} \}_{k=1,\ldots,n^{(c)}}, \)

\( \mathbf{e}^{(c,n^{(c)})}_{k} = \{ -\mathbf{h}^{(c,n^{(c)})}_{k} - \mathbf{g}^{(c,n^{(c)})}_{k} \} = \sum_{p=k}^{n^{(c)}} \left[ -\mathbf{h}^{(c,p)} - \mathbf{g}^{(c,p)} \right], \)

\( \mathbf{h}^{(c,p)} = \sum_{i,j=1}^{p} \mathbf{T}_i^{(c,p)} \mathbf{H}^{(c,p)} \mathbf{T}_j^{(c,p)^T} \mathbf{q}_i^{(c,p)} \mathbf{q}_j^{(c,p)}, \)

\( \mathbf{g}^{(c,p)} = \mathbf{m}^{(c,p)} \mathbf{g}^{[j_2]^T} \mathbf{T}_k^{(c,p)} \mathbf{r}_k^{(c,p)}, \)

\( \mathbf{t}^{(1,1)}_{dr} = \mathbf{t}^{(1,1)}_{dr}, \quad \mathbf{t}^{(1,1)}_{res} = \mathbf{t}^{(1,1)}_{res}, \)

\( \mathbf{s}^{(1,1)}_{f} = \{ \mathbf{s}^{(1,1)}_{f} \}, \quad \mathbf{s}^{(2,3)}_{f} = \{ \mathbf{s}^{(2,3)}_{f} \}. \)
A system of seven ordinary differential equations of second order was obtained which not only contains unknown components of the acceleration vectors \( \mathbf{q}^{(1,1)} \) and \( \mathbf{q}^{(2,3)} \), but also unknown components of the vector \( \mathbf{f}_S \). All of these unknown values were determined by using the Gauss elimination method. From the system presented, a system of four differential equations of second order, related to determining the components of the acceleration vectors \( \mathbf{q}^{(1,1)} \) and \( \mathbf{q}^{(2,3)} \), was eliminated and solved by the Newmark method with iterative procedure. As was mentioned, in order to compute the values of friction torques in the revolute joints \( R \), in each integrating step of the equations of motion additional calculations were performed using the recursive Newton-Euler algorithm and, as a result, the reaction force \( \mathbf{f}_{R(c,p)}^{(c,p)} \) and the reaction torque \( \mathbf{n}_{R(c,p)}^{(c,p)} \), acting on link \((c, p)\) by link \((c, p-1)\) in the revolute joint \((c, p)\), were determined. In the model of the revolute joint, which was assumed in this work, it was taken into account that the bearing liner is a part of link \((c, p-1)\), whereas the journal is a part of link \((c, p)\). The journal adjoins with the bearing liner by means of friction surfaces, i.e. revolute surfaces \( A \) and \( B \), and face surface \( C \) (Fig. 3). For the kinematic chains considered here it can be assumed that the origins of the local coordinate systems attached to the suitable links are located on the axes of the revolute joints \( R \) in half-distance \( \bar{a}^{(c,p)} \) defining the spacing of revolute friction surfaces \( A \) and \( B \) of the journals.

The values of the friction torques in the revolute joints \( R \) were calculated according to the following formula:

\[
\mathbf{f}_f^{(c,p)} = \mathbf{f}_{f,A}^{(c,p)} + \mathbf{f}_{f,B}^{(c,p)} + \mathbf{f}_{f,C}^{(c,p)}.
\]  (3)
The values of the friction torques acting on revolute friction surfaces $A$ and $B$ of the journals were described by the formulas:

$$
\tau_{f,A}^{(c,p)} = \frac{1}{2} \mu_{A}^{(c,p)} \mathbf{\gamma}_{A}^{(c,p)} \mathbf{D}_{A}, \\
\tau_{f,B}^{(c,p)} = \frac{1}{2} \mu_{B}^{(c,p)} \mathbf{\gamma}_{B}^{(c,p)} \mathbf{D}_{B}, \\
$$

whereas the value of the friction torque acting on face friction surface $C$ of the journal was expressed by means of the formula defined in paper [14]:

$$
\tau_{f,C}^{(c,p)} = \frac{1}{3} \mu_{C}^{(c,p)} \mathbf{\gamma}_{C}^{(c,p)} \mathbf{D}_{C} - \mathbf{a}^{(c,p)} - \mathbf{a}^{(c,p)}, \\
$$

The authors modelled the friction in the revolute joints $R$ using the Dahl model as described by means of the differential equation of first order which allows to analyse both phases of pre-sliding displacements (occurring after the change of signs of relative velocities in these joints – they could be understood as phases of static friction) and phases of motion (phases of kinetic friction). This model, presented by Dahl in succession in papers [15, 16, 17], was the first attempt to describe friction behaviour in the pre-sliding regime which exists before the normal slip occurs. Already in 1899, Stevens [18], with the help of an interferometer, found that elastic displacement exists in this regime. Since then many researchers, e.g. Courtney-Pratt and Eisner, who are authors of a classical paper in this field [19], also found similar results. Dahl presented his model after observing experimental data showed that for small displacements at the start of motion the friction force grows from zero to a steady-state value. When the direction of motion was reversed, again there was a finite regime in which the force grew to a new steady-state value. This resulted in hysteresis behaviour. Dahl developed his continuous “friction force–displacement” relationship on the base of the analogy with “stress–strain” dependence as presented, e.g. by Ramberg and Osgood in a known paper [20]. According to this theory, the transition from static friction to kinetic friction corresponds to the transition from elastic to plastic deformation in ductile materials. For large displacements the Dahl model is equivalent to the Coulomb friction model with a lag in the friction force change incorporated when relative velocity reverses sign.

For the requirements of the method proposed here, the values of instantaneous friction forces in the differential equation of the first order offered by Dahl were replaced by instantaneous friction coefficients and, as a result, the following form of this equation was obtained:

$$
\frac{d\theta_{a}^{(c,p)}}{dt} = \sigma_{a}^{(c,p)} \left( 1 - \frac{\mu_{a}^{(c,p)}}{\mu_{0,a}} \sgn \left( \psi_{a}^{(c,p)} \right) \right) \sgn \left( 1 - \frac{\mu_{a}^{(c,p)}}{\mu_{0,a}} \sgn \left( \psi_{a}^{(c,p)} \right) \right) for a \in \{A, B, C\} \\
$$
where $t$ – time,
$\mu_{\alpha (c,p)}$ – instantaneous friction coefficient in joint $(c,p)$,
$\mu_{\alpha (c,p)}^\text{inc}$ – maximum value of the instantaneous friction coefficient in joint $(c,p)$,
$\sigma$ – contact stiffness coefficient (it can be interpreted as the slope of the $\mu_{\alpha (c,p)}^\text{inc} (\psi^{(c,p)})$ curve obtained as a result of solving the equation presented, at $\mu_{\alpha (c,p)}^\text{inc} = 0$),
$i$ – parameter influencing the shape of the $\mu_{\alpha (c,p)}^\text{inc} (\psi^{(c,p)})$ curve.

Furthermore, in accordance with common practice, it was assumed that $i = 1$. After solving the equation presented here, the following formulas determining the courses $\mu_{\alpha (c,p)}^\text{inc} (\psi^{(c,p)})$ for $\alpha \in \{A,B,C\}$ (Fig. 4) were obtained:

$$
\mu_{\alpha (c,p)}^\text{inc} = \begin{cases} 
\left( \mu_{\alpha,\text{inc}}^{(c,p)} - \mu_{\alpha,0}^{(c,p)} \right) e^{\frac{\sigma}{\mu_{\alpha,0}^{(c,p)}} \left( \psi^{(c,p)} - \psi_{\text{inc}}^{(c,p)} \right)} + \mu_{\alpha,0}^{(c,p)} & \text{for } \psi^{(c,p)} \geq 0 \\
\left( \mu_{\alpha,\text{dec}}^{(c,p)} + \mu_{\alpha,0}^{(c,p)} \right) e^{\frac{\sigma}{\mu_{\alpha,0}^{(c,p)}} \left( \psi^{(c,p)} - \psi_{\text{dec}}^{(c,p)} \right)} - \mu_{\alpha,0}^{(c,p)} & \text{for } \psi^{(c,p)} < 0 
\end{cases}
$$

where $\mu_{\alpha,\text{inc}}^{(c,p)}$, $\psi_{\text{inc}}^{(c,p)}$, $\mu_{\alpha,\text{dec}}^{(c,p)}$, $\psi_{\text{dec}}^{(c,p)}$ are coordinates of points being the origins of increasing and decreasing curves, respectively.

**Figure 4.** Example of the “instantaneous friction coefficient–displacement” (and, more closely, “instantaneous friction coefficient–joint coordinate”) relationship.
The procedure used in the method, including calculations carried out in the scope of static and dynamic analysis of both open-loop kinematic chains, is presented in Fig. 5 in the form of suitable algorithms.

**Figure 5.** Algorithms of solving the equations of static and dynamic analysis of the system considered here.
5 NUMERICAL CALCULATION RESULTS

The assumed dimensions of the links and journals of the mechanism are presented in Tables 1 and 2.

Table 1. Parameters of the links.

<table>
<thead>
<tr>
<th></th>
<th>link (1,1)</th>
<th>link (2,1)</th>
<th>link (2,2)</th>
<th>link (2,3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$r^{(c,p)}$ [m]</td>
<td>0.2</td>
<td>0.2</td>
<td>0.1</td>
<td>0.3</td>
</tr>
<tr>
<td>$d^{(c,p)}$ [m]</td>
<td>0.028</td>
<td>0.028</td>
<td>0.028</td>
<td>0.028</td>
</tr>
</tbody>
</table>

Table 2. Parameters of the journals.

<table>
<thead>
<tr>
<th></th>
<th>joint (1,1)</th>
<th>joint (2,2)</th>
<th>joint (2,3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\overline{d}^{(c,p)}$ [m]</td>
<td>0.03</td>
<td>0.03</td>
<td>0.03</td>
</tr>
<tr>
<td>$d^{(c,p)}_A$ [m]</td>
<td>0.01</td>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>$d^{(c,p)}_B$ [m]</td>
<td>0.01</td>
<td>0.01</td>
<td>0.01</td>
</tr>
<tr>
<td>$d^{(c,p)}_C$ [m]</td>
<td>0.015</td>
<td>0.015</td>
<td>0.015</td>
</tr>
<tr>
<td>$\mu_A^{(c,p)}$</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
</tr>
<tr>
<td>$\mu_B^{(c,p)}$</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
</tr>
<tr>
<td>$\mu_C^{(c,p)}$</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
</tr>
</tbody>
</table>

The courses of the value of the driving torque $t^{(1,1)}_{dr}$ and the reduced resistance torque $t^{(1,1)}_{res}$, assumed respectively as a function of time $t$ and joint velocity $\psi^{(1,1)}$, are presented in Figs. 6a and 6b. These courses are to allow the angular velocity of the driving link to oscillate around a selected value $\psi^{(1,1)}_0$. The velocities of the other links will fix themselves around suitable values adequate to the assumed value of the angular velocity of the driving link. In the case considered here, the following parameters were taken into account: $t^{(1,1)}_{dr,0}=10\text{Nm}$, $t_{st}=2\text{s}$ and $\psi^{(1,1)}_0=9\text{rad/s}$. 
The obtained courses of the reaction forces and torques in the joints of the mechanism as functions of time are shown in Fig. 7 – when friction in the joints was either taken into account or omitted.
The analysis of the plots presented here allows to state that friction significantly slows down the
courses (the values of the quantities presented in the case of joints with friction are achieved
later than in the case of ideal joints). The numerical experiments proved that only three
iterations were required, as further increasing the accuracy of the calculations did not cause
perceptible differences in the results obtained. Moreover, these experiments did not confirm the
need to use the constraint stabilisation method in order to increase the accuracy of the numerical
calculations.

6 CONCLUSIONS

Dynamic analysis of a selected RSRRR spatial linkage mechanism with friction in revolute
joints R is presented in the article. This mechanism can be qualified to a specific class of spatial
one-dof linkage mechanisms containing one spherical joint in their structure. The proposed
method of dynamic analysis assumes a division of the mechanism in the place of the spherical
joint into two open-loop kinematic chains. Thus an effective approach, which is mainly used for
dynamic analysis of manipulators that are in fact open-loop kinematic chains, can also be used
in the dynamic analysis of such a mechanism. The effective recursive Newton-Euler algorithm,
also taken from robotics, was used to define the values of friction torques. The method proposed
here has general significance and could be used to analyse other spatial linkage mechanisms of
this class.

REFERENCES

and solution of the equations of motion for complex mechanisms. In E. Bautista,
J. Garcia-Lomas, A. Navarro (Eds). Proc. of 7-th World Congress on The Theory of

Basic methods, Allyn and Bacon, 1989.


Nonlinear analysis of the cable-pulley interaction

Michal Hajžman∗, Radek Bulín∗, Pavel Polach∗

∗ European Centre of Excellence, NTIS – New Technologies for Information Society
Faculty of Applied Sciences
University of West Bohemia
Univerzitní 8, 306 14 Plzeň, Czech Republic
mhajzman@kme.zcu.cz, [rbulin, ppolach]@ntis.zcu.cz

ABSTRACT
This paper deals with the modelling methodology for the mechanical systems composed of cables and pulleys or sheaves. The Absolute Nodal Coordinate Formulation was chosen as a suitable approach that can allow to consider detailed interaction of a cable and a pulley with its nonlinear behaviour. An in-house modelling tool in the MATLAB system was created based on the proposed modelling methodology. Its performance was tested on a simple chosen cable-pulley system and on a real measured system with the kinematic excitation by the linear motor.

Keywords: Contact, Friction, ANCF, Experiment, Numerical simulation.

1 INTRODUCTION
The usage of cables, wires and fibres in various machines can be found in the real world and therefore it is necessary to have proper and effective modelling tools for sake of dynamic analysis and e.g. control synthesis. Modelling approaches suitable for cables can be divided into several groups based on complexity.

The simplest way how to incorporate cables in the equations of motion of a mechanism is the force representation of a cable (e.g. [1, 2]). It is supposed that the mass of cables is small to such an extent comparing to the other moving parts that the inertia of cables is negligible with respect to the other parts. The cable is represented by the force dependent on the cable deformation and its stiffness and damping properties. A variable length of the cable due to wiring can be easily described using the force approach. A more accurate approach is based on the representation of the cable by a point-mass model (e.g. [3]). The cable can be considered either flexible or rigid. It has the advantage of a lumped point-mass model. The point masses can be connected by forces or constraints. Wiring of a cable can be also simulated and a detailed model of a wiring mechanism can be observed. In order to represent also bending behaviour of cables their discretization using the finite segment method [4] or so called rigid finite elements [5] is possible. Standard multibody codes (MSC.ADAMS, SIMPACK) can be used for this purpose. Other more complex approaches can utilize nonlinear 3D finite elements [6] or can employ ANCF (Absolute Nodal Coordinate Formulation) elements [4]. ANCF is capable to be used in the modelling of mechanical systems considering cables or wires as was demonstrated in [7] or [8].

This paper deals with a detailed planar analysis of the interaction between a rotating pulley (a sheave) and a cable, which is modelled using ANCF approach. An in-house modelling tool in the MATLAB system was created based on the proposed modelling methodology. Its performance was tested on a simple chosen cable-pulley system and on a real measured system with the kinematic excitation by the linear motor.

2 CABLE MODELLING
A planar ANCF beam element of length $l$ with two nodes is employed (see Figure 1). Global position $\textbf{r} = [r_1, r_2]^T$ of an arbitrary beam point determined by parameter $p$ can be written as

$$\textbf{r}(p) = S(p)\textbf{e}, \quad \textbf{e} = [e_1, e_2, \ldots, e_8]^T,$$  

(1)
where $S \in \mathbb{R}^{2,8}$ is a global shape function matrix, $\mathbf{e}$ is a vector of element nodal coordinates and $p \in (0, l)$ is a parameter of a curve.

\[ E_k = \frac{1}{2} \int_0^l \rho A \epsilon^T \dot{\mathbf{r}} \, dp = \frac{1}{2} \dot{\mathbf{e}}^T \int_0^l \rho A S^T \dot{S} \, dp \dot{\mathbf{e}} = \frac{1}{2} \dot{\mathbf{e}}^T \mathbf{M}_e \dot{\mathbf{e}}, \]  

(2)

where element mass matrix

\[
\mathbf{M}_e = \frac{\rho A l}{420} \begin{bmatrix}
156\mathbf{I} & 22\mathbf{I} & 54\mathbf{I} & -13\mathbf{I} \\
4l^2\mathbf{I} & 13\mathbf{I} & -3l^2\mathbf{I} \\
156\mathbf{I} & -22\mathbf{I} & 4l^2\mathbf{I}
\end{bmatrix}_{\text{sym.}}
\]  

(3)

is a constant matrix in case of this formulation.

Strain energy $E_p$ of the element is used for the derivation of elastic forces in the ANCF beam model and the form of an adopted elasticity model determines the complexity of the whole model. In [9] and [10], there are several approaches, which employ the separation of the strain energy of longitudinal deformation $E_{pl}$ and the strain energy of transverse (bending) deformation $E_{pt}$ as

\[ E_p = E_{pl} + E_{pt} = \frac{1}{2} \int_0^l EA \epsilon^2 \, dp + \int_0^l EI \kappa^2 \, dp, \]  

(4)

where $E$ is Young modulus, $A$ is the area of the cross-section and $I$ is the second moment of the area about a transverse axis. The possible models are then classified according to the expressions for longitudinal strain $\epsilon$ and curvature $\kappa$. General expressions for these quantities are

\[ \epsilon = \frac{1}{2} (\mathbf{r}'^T \mathbf{r}' - 1), \quad \kappa = \left| \frac{d^2 \mathbf{r}}{ds^2} \right|, \]  

(5)

where $s$ is the parameter of a curve. Berzeri and Shabana [9] introduced several suitable models for both longitudinal and transverse elastic forces, whereas the formulation denoted L2T2 is employed in this paper in order to investigate the cable-pulley interaction.

The whole model of the ANCF planar beam element [9] is of the form

\[ \mathbf{M}_e \ddot{\mathbf{e}} + \mathbf{K}_e(\mathbf{e}) \mathbf{e} = \mathbf{Q}_{ek}, \]  

(6)

and is characterized by constant mass matrix $\mathbf{M}_e$, strongly nonlinear stiffness matrix $\mathbf{K}_e(\mathbf{e})$ derived using the strain energy and by vector of external forces $\mathbf{Q}_{ek}$. The assembling of a discretized
flexible body (i.e. fibre, cable) model is straightforward and can be extended by a suitable model of viscous forces $B(\dot{q}, q)$

$$M\ddot{q} + B(\dot{q}, q)\dot{q} + K(q)q = Q,$$  

(7)

where $q$ is the vector of all elastic coordinates of the fibre. This model can be combined with the models of other flexible or rigid bodies and with the model of kinematic joints using a standard way.

### 3 CABLE-PULLEY INTERACTION

The pulley is modelled as a rigid body with one degree of freedom (rotation), the fibre is modelled as a deformable body and it is discretized using the ANCF method to $n$ elements. During the interaction of these two bodies, the contact forces arise not only in element nodes (Figure 2). Therefore $k$ equally spaced points are determined on each element. The contact forces are then evaluated for each of these point. Let us have element $e$ ($e = 1, \ldots, n$) and point $i$ ($i = 1, \ldots, k$). In each time step, the relative penetration $\delta_{ei}$, penetration velocity $\dot{\delta}_{ei}$ and relative tangential velocity vector $v_{ei}$ between the pulley and the cable point is determined. Than the normal and friction forces can be evaluated.

![Figure 2. Interaction of a cable and a pulley.](image)

The best-known and very popular normal contact force formula was introduced by Hertz [11]. This model is based on the theory of elasticity and describes a force between two perfectly flexible solids with frictionless surfaces. The Hertz law can be expressed as

$$F_{Nej} = K\delta_{ei}^n,$$  

(8)

where $K$ is the contact stiffness and $n$ is the positive exponent, which reflects the shape of contact bodies. This basic contact force model does not consider the dissipation of energy during contact, therefore more complex models were developed [12]. Hunt-Crossley’s model of the normal contact force [11]

$$F_{Nej} = K\delta_{ei}^n(1 + D\dot{\delta}_{ei})$$  

(9)
was proposed as a more complex modelling approach. It is based on a simple dissipation model, where the damping coefficient is dependent on the penetration. The friction forces were also added to the interaction model and the model performance was tested with respect to various parameters of the friction model. Friction force model is based on Threlfall model \[11\]

\[ F_T = c_f F_N \left(1 - e^{-3(v_T/v_r)}\right). \] (10)

The contact forces are evaluated in each point and then used for the calculation of vector of element contact forces.

Now, the vector of the contact forces acting on element \(e\) caused by interaction of pulley and point \(i\) has the form

\[ \mathbf{f}_{Cei} = S^T(p_i) \begin{bmatrix} F_{Nei} \\ F_{Tei} \end{bmatrix}, \] (11)

where \(p_i\) is the parameter, which determines the position of point \(i\) on element \(e\).

The cable model can be combined with the models of other flexible or rigid bodies and with the model of kinematic joints. Due to the usage of absolute displacements as nodal coordinates it is easy to define the kinematic constraint equations such as revolute or translational joints. The constraints between the chosen coordinates can be written using the vector notation

\[ \Phi(q, t) = 0 \] (12)

and for the use in equations of motion it must be differentiated to obtain the Jacobian matrix

\[ \Phi_q = \frac{\partial \Phi}{\partial q} = \begin{bmatrix} \frac{\partial \Phi_i}{\partial q_j} \end{bmatrix}, \quad i = 1, 2, \ldots, m, \quad j = 1, 2, \ldots, n, \] (13)

where \(m\) is the number of constraints and \(n\) is the number of dependent coordinates. After the introduction of vector of Lagrange multipliers \(\lambda\) the whole system of equations, which fully describe the problem of the coupled system of flexible and rigid bodies, can be written in the matrix form

\[ \begin{bmatrix} M & \Phi_q^T \\ \Phi_q & 0 \end{bmatrix} \begin{bmatrix} \ddot{q} \\ -\lambda \end{bmatrix} = \begin{bmatrix} g(q, \dot{q}, t) \\ \gamma(q, \dot{q}, t) \end{bmatrix} = \begin{bmatrix} Q_k - B(q, \dot{q})\dot{q} - K(q)q \\ \gamma(q, \dot{q}, t) \end{bmatrix}. \] (14)

Generally, vector \(g(q, \dot{q}, t)\) contains the sum of all generalized, potential and dissipative forces and appropriate other derivatives of a kinetic energy, while vector \(\gamma(q, \dot{q}, t)\) follows from the differentiation of the constraint equations (see [4]). The approaches to the numerical solution of equation (14) used practically in this paper are summarized in [13].

4 SIMPLE APPLICATION

The simple mechanical system consisting of a cable and a pulley (see Figure 3) with chosen parameters collected in Table 1 was used for the testing of an in-house simulation tool implemented in the MATLAB system based on the presented modelling methodology.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fibre radius</td>
<td>(r_f = 0.001) m</td>
</tr>
<tr>
<td>Sheave radius</td>
<td>(r_p = 0.05) m</td>
</tr>
<tr>
<td>Fibre length</td>
<td>(l_f = 1) m</td>
</tr>
<tr>
<td>Modulus</td>
<td>(E_f = 1\times10^9) MPa</td>
</tr>
<tr>
<td>Weight 1 (left)</td>
<td>(m_1 = 1) kg</td>
</tr>
<tr>
<td>Weight 2 (right)</td>
<td>(m = 10) kg</td>
</tr>
</tbody>
</table>

The important quantity for the analysis of the performance of the implemented model was the rotation (sheave angle) of the pulley during the interaction with the cable. Time history of the pulley motion for various cable models characterized by different number of finite elements is in Figure 4. It can be seen that for the global motion the differences could not be visible and therefore Figure 5 shows the difference between the signal.

557
Analogous analysis was performed in order to investigate the effect of the number of element points for the contact force evaluation. Time history of the pulley motion for various cable models characterized by different number of points for contact force evaluation is in Figure 6. It can be noticed that even for the small number of points the global dynamic response is very similar. The differences between time histories are shown in Figure 7.

Details about contact forces in normal and tangential direction (indexes N and T) can be seen in Figure 8, where the time histories of these forces are drawn for chosen nodes in the mutual contact with the pulley. Moreover, the analysis is performed for two different normal force models. It is clear that the Hertz model, which doesn’t contain dissipation, will exhibit more oscillating behaviour.
Figure 5. Difference between time history of rotation of the pulley for the simple testing model — effect of the number of ANCF elements.

Figure 6. Rotation of the pulley for the simple testing model — effect of the number \( k \) of point for the contact forces evaluation.

5 COMPARISON OF NUMERICAL SIMULATIONS WITH MEASURED RESULTS

This section deals with the comparison of the results obtained by the numerical simulations and by experiments. Experimental measurements focused on the investigation of the fibre behaviour were performed on an assembled weigh-fibre-pulley-drive mechanical system (see Figure 9), which was described e.g. in [14, 15]. A carbon fibre with a silicone coating is driven with one drive and it is led over a pulley. The fibre length is 1.82 meters (fibre weight is 4.95 grams), the pulley diameter is 80 millimetres. At the drive the fibre is fixed on a force gauge. In the other end of the fibre there is a prism-shaped steel weight (weight of 5.035 kilograms in this case), which moves in
a prismatic linkage on an inclined plane. The angle of inclination of the inclined plane can be changed for different measurements. In the case presented in this paper the weight position angle $\alpha$ is 30 degrees and the pulley-fibre angle $\phi$ is 150 degrees and it is changing during the motion. Drive excitation signals can be of a rectangular, a trapezoidal and a quasi-sinusoidal shape and there is a possibility of variation of a signal rate. The amplitudes of the drive displacements are up to 90 millimetres.

The particular signal defining the motion of the motor is in Figure 10 together with the measured position of the weight with respect to time.

The measured motion of the motor served as an input signal (kinematic excitation) for the numerical simulations. Chosen calculated dynamic response of the weight using complex cable-pulley
Figure 9. Scheme of the testing mechanical system composed of a motor, a pulley, a weight and a cable.

interaction based on ANCF beam elements is shown in Figure 11. It can be concluded that without any effort to tune the model parameters, the correspondence of the measured and calculated results is relatively sufficient. Several parameter studies were performed for the same configuration of the experiment with various parameters of contact models and the conclusions are almost the same.

Figure 10. Measured motion of the motor and the weight during the experiment.
Further, the effect of different cable behaviour in tension and in press was implemented in the model. Obtained results for the studied rate of motor motion didn’t show significant differences between the models.

In order to document the complex behaviour of the cable during the motion, Figure 12 shows time history of the transversal motion of the chosen node between the pulley and the motor. This motion characterized transversal vibration of the cable which could be the problem during the motion control of accurate mechanisms.

**Figure 11.** Calculated and measured motion of the weight.

**Figure 12.** Calculated transversal vibration in a chosen node of the cable between the pulley and the motor.
6 CONCLUSIONS

The beam based on the Absolute Nodal Coordinate Formulation was employed for the modelling of cables, which could be in interaction with pulleys or sheaves. Detailed studies were performed at first on the chosen simple system composed of one pulley, one cable and two masses without any kinematic excitation for sake of the validation of created in-house software tool in the MATLAB system.

Further the cable-pulley interaction was studied on the weight-fibre-pulley-motor system with prescribed kinematic excitation of the motor. Similar problem was also addressed e.g. in [8] but the contribution of this paper is in the comparison with experimental results measured on the real mechanical system.

Future work will be focused on the possibilities of different behaviour in tension and in press and on the accurate modelling of tangential forces respecting stick-slip motion.

The paper has originated in the framework of solving No. 15-20134S project of the Czech Science Foundation entitled “Multi-level Light Mechanisms with Active Structures”.

REFERENCES


On the Frictional Contacts in Multibody System Dynamics

Filipe Marques*, Paulo Flores*, Hamid M. Lankarani#

*Department of Mechanical Engineering
School of Engineering, University of Minho
Campus de Azurém, 4804-533 Guimarães - PT
filipe62055@gmail.com, pflores@dem.uminho.pt

#Wichita State University
1845 N. Fairmount Street
Wichita, KS 67260-0133 USA
hamid.lankarani@wichita.edu

ABSTRACT

A comprehensive analysis on the use of different friction force models on the dynamic simulations of multibody systems is presented in this study. In this context, the most relevant approaches for dealing with friction available in the literature are revisited, classified into the statics and dynamics models. In this process, the main limitations and implications of the friction force models are briefly described. The dynamic response of a single-mass one degree-of-freedom system is utilized to analyze and compare the various friction laws. The results obtained suggest that the prediction of the dynamic behavior of multibody systems strongly depends on the selection of the appropriate friction model as well as frictional parameters.

Keywords: Friction, Multibody dynamics, Contact forces, Stick-slip.

1 INTRODUCTION

Friction is generally related to the resistance to the relative motion between different surfaces in contact. There are different phenomena associated with friction, such as the dry friction and the lubrication friction. Friction is a highly complex phenomenon, which occurs in all mechanical systems. In some cases, the presence of friction is desirable, e.g. while walking or braking a car. However, friction has in general provided resistance and could have negative effects.

Due to its importance, friction has been studied over the centuries. One of the first works appeared in the 16th century resulting from Leonardo da Vinci’s work, who stated that the friction force is proportional to normal load, opposes to the motion, and is independent of the contact area. Later, Amontons corroborated with da Vinci. Coulomb stated that friction was independent of velocity magnitude, and developed the first friction model [1].

During the relative movement between different bodies, when the relative velocity reduces, the friction force increases, which may lead to the sticking of contacting surfaces. This phenomenon is the so-called “stick-slip” [2], which is overcome when the applied force is higher than the static friction force, and consequently the bodies start sliding. Rabinowicz [3] investigated the essence of the static and the kinetic coefficients of friction, and experimentally demonstrated that the coefficient of friction can be described as a function of displacement, with a maximum value for small displacements.

The necessary force to initiate the motion and to overcome the static friction is called the brake-away force, which corresponds the maximum friction force [3]. The overcoming of this force defines the boundary between the sticking and sliding regimes. It is known that when an external tangential force is applied between two contacting bodies, a small displacement occurs, which resembles a spring behavior. This phenomenon of a small motion in elastic range, when the applied force is less than the break-away force, is often referred as pre-sliding displacement.

Strubeck [4] proposed a steady state curve for friction force as a function of relative sliding velocity. This dependence contradicts the Coulomb's law. However, Coulomb friction is still utilized due to its simplicity and is the basis for many friction models available on the literature [5-8]. Most of the modified models have been developed to deal with viscous friction, stiction, frictional lag, amongst others.
Hence, the main goal of this work is to present and compare several friction force models that can be utilized in the context of multibody systems formulations. In the sequel of this process, the most relevant static and dynamic friction models are briefly described. Subsequently, a simple numerical application example is considered to illustrate the dynamic response of the using of different friction approaches.

2 STATIC FRICTION MODELS

This section includes several “static” friction force models frequently used in the simulations of multibody mechanical systems. It must be stated that most of these models present a discontinuity of friction force when the relative velocity is zero, which can cause difficulties in describing friction realistically.

2.1 Coulomb Friction

Coulomb presented the first friction model which states that the friction always opposes relative motion between contacting bodies and its magnitude is proportional to the normal contact force. This model depends on the relative velocity direction, except for zero velocity where the friction force is a multivalued function of the external tangential force. This model can described as [1]

\[
F = \begin{cases} 
    F_C \text{sgn}(\mathbf{v}) & \text{if } \|\mathbf{v}\| \neq 0 \\
    \min(\|\mathbf{F}_e\|, F_C) \text{sgn}(\mathbf{F}_e) & \text{if } \|\mathbf{v}\| = 0 
\end{cases}
\]  

(1)

where

\[
F_C = \mu_k \|\mathbf{F}_N\|
\]  

(2)

in which \(\mathbf{F}_N\) is the normal force, \(F_C\) is the magnitude of Coulomb friction, \(\mu_k\) is the kinetic coefficient of friction, \(\mathbf{F}_e\) is the external tangential force, and \(\mathbf{v}\) is the relative velocity of the contacting bodies. This model presents a dependence on the velocity by the signum function,

\[
\text{sgn}(\mathbf{v}) = \begin{cases} 
    \mathbf{v} & \text{if } \|\mathbf{v}\| \neq 0 \\
    \mathbf{0} & \text{if } \|\mathbf{v}\| = 0 
\end{cases}
\]  

(3)

where \(\mathbf{0}\) is a null vector with the same dimensions as \(\mathbf{v}\). Although this model is straightforward, it presents some difficulties since it does not specify a friction force at zero velocity. Thus, this velocity dependence can originate perturbations in the dynamic simulations system’s response. Nevertheless, the Coulomb friction law is used quite often to simulate friction behavior for the sake of simplicity, and since it only requires one input parameter; i.e. the coefficient of friction.

2.2 Coulomb Model with Stiction

Since the friction force at zero velocity is higher than the kinetic friction, several studies have presented the necessity of introducing a friction model which includes two different friction coefficients. This modified Coulomb approach has a similar behavior to Coulomb’s except in the vicinity of zero velocity. It is also a multivalued function, but can reach a higher friction force, and can be described as follows [8]

\[
F = \begin{cases} 
    F_C \text{sgn}(\mathbf{v}) & \text{if } \|\mathbf{v}\| \neq 0 \\
    \min(\|\mathbf{F}_e\|, F_S) \text{sgn}(\mathbf{F}_e) & \text{if } \|\mathbf{v}\| = 0 
\end{cases}
\]  

(4)

where

\[
F_S = \mu_s \|\mathbf{F}_N\|
\]  

(5)

in which \(F_S\) is the magnitude of static friction, and \(\mu_s\) is the static coefficient of friction which is higher than the kinetic, \(\mu_k\). Although this model considers stiction, in practice, it provides similar behavior compared to Coulomb’s law, with an oscillatory force for low velocities.
2.3 Coulomb Model with Viscous Friction

One of the most common modifications of Coulomb’s friction law deals with the introduction of viscous friction component. Considering a linear relationship between the relative velocity and the friction force related to the lubricant’s viscosity, the friction model can be written as [8]

\[
F = \begin{cases} 
F_v \text{sgn}(v) + F_v v & \text{if } \|v\| \neq 0 \\
\min(\|F_v\|, \|F_s\|) \text{sgn}(F_v) & \text{if } \|v\| = 0
\end{cases}
\]  

(6)

where \(F_v\) is the viscous friction coefficient.

2.4 Model with Stribeck Effect

In contrast with the Coulomb model with stiction, the Stribeck effect [4] ensures that the decrease from static to kinetic friction is a continuous process. Thus, the friction force during relative motion is expressed as a continuous function of velocity as

\[
F = \begin{cases} 
F(v) & \text{if } \|v\| \neq 0 \\
\min(\|F_v\|, \|F_s\|) \text{sgn}(F_v) & \text{if } \|v\| = 0
\end{cases}
\]  

(7)

where \(F(v)\) is an arbitrary function that depends on the relative velocity. With this model, the friction force diminishes when the relative motion is initiated. Bo and Pavelescu [9] introduced an exponential function as follows

\[
F(v) = F_v + (F_s - F_v) e^{-\frac{\|v\|}{\nu_s}} \text{sgn}(v) + F_v v
\]  

(8)

where \(\nu_s\) is the Stribeck velocity and \(\delta_v\) is a factor that relies on the geometry of the contacting surfaces, which is often considered 2 as suggested by Armstrong-Hélouvry [10]. This friction model takes into account the Coulomb, viscous, stiction, and Stribeck friction effects. However, it presents the same problem as the previous approaches at zero velocity.

2.5 Karnopp Model

Since the aforementioned models are multivalued functions for zero velocity, their static behavior cannot be captured during a simulation. To overcome this difficulty, Karnopp [11] proposed a model where the velocity is considered zero, for a specified range. Thus, when the velocities are within the interval, the system’s state can change and the model’s response will be the same as when the relative velocity is zero.

Karnopp model is usually used along the Coulomb model and can be expressed as

\[
F = \begin{cases} 
F(v) & \text{if } \|v\| > D_v \\
\min(\|F_v\|, \|F_s\|) \text{sgn}(F_v) & \text{if } \|v\| \leq D_v
\end{cases}
\]  

(9)

where \(D_v\) is the tolerance for zero velocity. It is important to select a suitable range of the null velocity. Nevertheless, this zero velocity interval does not comply with the real behavior of the contact.

2.6 Threlfall Model

In most of the static models described above, the friction force at zero velocity is multivalued, being evaluated as a function of the external tangential force. In order to simplify and ensure computational efficiency, several researchers have proposed alternative methods, which replace the discontinuity at zero velocity by a finite slope model. Thus, Threlfall [5] presented a model that avoids the discontinuity associated with the Coulomb’s law,
\[
\mathbf{F} = \begin{cases} 
F_C \left(1 - e^{-\frac{\|\mathbf{v}\|}{v_0}}\right) \text{sgn}(\mathbf{v}) & \text{if } \|\mathbf{v}\| \leq v_0 \\
F_C \text{sgn}(\mathbf{v}) & \text{if } \|\mathbf{v}\| > v_0
\end{cases}
\]

where \(v_0\) is a specified tolerance velocity. The resemblance with Coulomb friction law increases with the decreasing of this tolerance velocity.

### 2.7 Bengisu and Akay Model

Bengisu and Akay [12] proposed an approach capable of modelling the Strubeck effect, and it can be defined as

\[
\mathbf{F} = \begin{cases} 
\left(\frac{F_C}{v_0}\left(\|\mathbf{v}\| - v_0\right)^2 + F_s\right) \text{sgn}(\mathbf{v}) & \text{if } \|\mathbf{v}\| < v_0 \\
F_C + (F_s - F_C)\exp^{-\frac{v_0}{v_0}} \text{sgn}(\mathbf{v}) & \text{if } \|\mathbf{v}\| \geq v_0
\end{cases}
\]

in which \(\zeta\) should be a positive parameter representing the negative slope of the sliding state. When the slope at zero velocity is too large, a small step size is needed to correctly capture friction for low velocities, which will slow down the simulation. In addition, for velocities close to zero, the friction force will always be low irrelevant of the displacement.

### 2.8 Ambrósio Model

The above mentioned limitations associated with friction force’s discontinuity led Ambrósio [6] to propose a modified Coulomb’s friction law in which the friction force can be defined as

\[
\mathbf{F} = \begin{cases} 
0 & \text{if } \|\mathbf{v}\| \leq v_0 \\
\frac{\|\mathbf{v}\| - v_0}{v_1 - v_0} F_C \text{sgn}(\mathbf{v}) & \text{if } v_0 < \|\mathbf{v}\| < v_1 \\
F_C \text{sgn}(\mathbf{v}) & \text{if } \|\mathbf{v}\| \geq v_1
\end{cases}
\]

where \(v_0\) and \(v_1\) are the tolerances for the velocity. This approach prevents the friction force from changing direction when the relative velocity is close to zero. However, it does not describe the stick-slip motion.

### 2.9 Awrejcewicz Model

Awrejcewizc et al. [13] proposed a static friction model for dry contact which is dependent of both tangential force and relative velocity. This model is governed by four different equations, one for sliding, two for the transition from stick to slip, and one for sticking, as follows

\[
\mathbf{F} = \begin{cases} 
\mathbf{F}(\mathbf{v}) & \text{if } \|\mathbf{v}\| > \varepsilon \\
F_s \text{sgn}(\mathbf{F}_e) & \text{if } \|\mathbf{v}\| \leq \varepsilon \land \|\mathbf{F}_s\| > F_s \land \mathbf{F}_e \cdot \mathbf{v} \geq 0 \\
(2A-1)F_s \text{sgn}(\mathbf{v}) & \text{if } \|\mathbf{v}\| \leq \varepsilon \land \|\mathbf{F}_s\| > F_s \land \mathbf{F}_e \cdot \mathbf{v} < 0 \\
A(-\mathbf{F}_e + F_s \text{sgn}(\mathbf{v})) + \mathbf{F}_e & \text{if } \|\mathbf{v}\| \leq \varepsilon \land \|\mathbf{F}_s\| \leq F_s
\end{cases}
\]

in which

\[
A = \frac{\|\mathbf{v}\|^2}{\varepsilon^2} \left(3 - 2\frac{\|\mathbf{v}\|}{\varepsilon}\right)
\]

where \(\varepsilon\) is a velocity tolerance, and \(\mathbf{F}(\mathbf{v})\) is an arbitrary friction function for sliding which depends on the velocity. The tolerance velocity defines the limit for sliding state. Below this tolerance, the friction force is also calculated as a function of the external tangential force.
3 Dynamic Friction Models

This section includes some of the most relevant dynamic friction models. As described earlier, in general, the static friction approaches have limitations in capturing some friction phenomena, such as pre-sliding displacement or frictional lag. Thus, better alternatives should be discussed, namely the available dynamic friction models, also named state variable models. In a simple manner, the dynamic models use an extra state variable used together with the velocity to calculate the friction force.

3.1 Dahl Model

The Dahl friction model [14] was developed with the aim of describing the friction behavior of ball bearings. The basis of this solution is an analogy with the classical stress-strain curve of materials. Dahl observed that in brittle materials, the difference between the stiction and Coulomb friction is difficult to capture. Ductile materials, however, are more probable of having the stiction behavior and then decrease the stress until Coulomb friction is reached. Moreover, it was shown that, the friction force is dependent on relative velocity and displacement. Dahl model states that when the contacting surfaces are subjected to stress, the friction force increases until rupture occurs. In this context, the stress-strain curve can be described by a differential equation as

\[
\frac{dF}{dx} = \sigma \left[ 1 - \frac{F}{F_C} \operatorname{sgn}(v) \right] \operatorname{sgn}\left(1 - \frac{F}{F_C} \operatorname{sgn}(v)\right)
\]

where \( F \) denotes the friction force, \( x \) is the displacement, \( F_C \) is the Coulomb friction, \( \sigma \) represents the stiffness coefficient and \( \alpha \) is a parameter that defines the shape of the material curve. This parameter depends on the material, and usually varies between 0 and 1 for brittle materials, and is higher than 1 for ductile materials. From the analysis of (14), when \( F \) tends to \( F_C \), the derivative tends to zero, so it can be concluded that the magnitude of the friction force does not exceed \( F_C \).

Equation (14) can be modified to a time derivative, and generalized for the 3D case. The most common value for \( \alpha \) is 1, yielding the Dankowicz model as [15],

\[
\frac{dF}{dt} = \sigma \left(1 - \frac{F \cdot \operatorname{sgn}(v)}{F_C}\right)v
\]

Introducing the state variable \( z \), and assuming that \( F=\sigma z \), Eq. (15) can be written as

\[
\frac{dz}{dt} = \left(1 - \frac{\sigma}{F_C} \cdot z \cdot \operatorname{sgn}(v)\right)v
\]

It can be observed from (16) that when the system reaches the steady state, the friction force is

\[
F = F_C \operatorname{sgn}(v)
\]

which is in fact the Coulomb friction model.

It must be highlighted that the Dahl model is not capable of capturing the Stribeck effect and stiction, since it is based on the dry Coulomb friction model with the introduction of pre-sliding displacement through a new state variable, eliminating the discontinuity at zero velocity.

3.2 Reset Integrator Model

Haessig and Friedland [16] proposed an evolution of the Dahl model, which considers that the friction force is originated by the elastic and plastic deformations of the surface asperities. Each contact is modeled as a bond between two bristles. The reset integrator model does not allow for the bond to break, which means that when the strain of a connection increases until reaching the rupture point, the model ensures that it is kept constant. This model uses the average of bristle deflection \( z \) to determine the strain in the bond and to account the stiction, as
\[
\frac{dz}{dt} = \begin{cases} 
0 & \text{if } \|z\| \geq z_0 \land z \cdot v > 0 \\
v & \text{otherwise}
\end{cases}
\]  
(18)

Similar to other friction models, the reset integrator model is also composed of two state equations, one for sticking and another for sliding. The transition between those two phases occurs when the deflection reaches its maximum value \(z_0\). This friction force can then be defined as follows

\[
F = \begin{cases} 
\sigma_0(v)(1+a)z + \sigma_1 \frac{dz}{dt} & \text{if } \|z\| < z_0 \\
\sigma_0(v)z_0 \text{sgn}(z) & \text{if } \|z\| \geq z_0
\end{cases}
\]  
(19)

where \(\sigma_1 dz/dt\) is the damping term that introduces some physical meaning by having damping oscillations and viscous friction effects, \(a\) denotes the coefficient pertaining to the stiction, and \(\sigma_0(v)\) is the contact stiffness. This friction force model has a discontinuity when the analysis changes between sticking and sliding situations.

### 3.3 LuGre Model

The LuGre model was proposed by Canudas de Wit et al. [17] and can be considered as a derivation from the Dahl model [14]. This model is capable of capturing the Strubeck and stiction effects. In a simple way, this model considers friction as the result of the interactions of the surfaces bristles, it is regards to the average of the bristle deflection. When a force is applied, the bristles start to deform with spring behavior during the sticking phase. Then if the force is sufficiently large, the bodies start to slip. The model follows as

\[
\frac{dz}{dt} = \left(1 - \frac{\sigma_0}{g(v)}z \cdot \text{sgn}(v)\right)v
\]  
(20)

\[
F = \sigma_0 z + \sigma_1 \frac{dz}{dt} + f(v)
\]  
(21)

where \(\sigma_0\) is the stiffness of the bristles, \(\sigma_1(v)\) is the damping of the bristles which can be set at constant, or can be a function of velocity, \(f(v)\) is an arbitrary function that describes the viscous effect and \(g(v)\) is an arbitrary function that accounts for the Strubeck effect as

\[
g(v) = F_C + (F_S - F_C)e^{\frac{\|v\|}{v_S}}
\]  
(22)

where \(F_C\) is the Coulomb friction, \(F_S\) is the static friction and \(v_S\) is the characteristic velocity of the Strubeck friction [10]. For \(f(v)\), typically a linear viscous friction is considered, that is

\[
f(v) = \sigma_2 v
\]  
(23)

The effect of this term is quite important when there is a fluid lubricant or when the relative velocities are high.

For a constant velocity, that is, when the system reaches the steady state \((dz/dt=0)\), the expression to the friction force can be reduced to

\[
F = g(v)\text{sgn}(v) + f(v)
\]  
(24)

Thus, considering the equations (22) and (23), it is possible to conclude that the steady-state values of the LuGre model agrees with the static model defined by Eq.(8).

### 3.4 Elasto-Plastic Model

The elasto-plastic model was developed by Dupont et al [18], and it is based on the division of the body displacement into two different components, the elastic and plastic displacement. When the bodies are sticking, the plastic displacement remains constant, while during the sliding phase, the elastic displacement is constant. The friction force can be calculated in a similar way as compared to the case of LuGre model [17],
\[ F = \sigma_0 z + \sigma_1 \frac{dz}{dt} + \sigma_2 v \]  
(25)

where \( \sigma_0 \) is the contact stiffness, \( \sigma_1 \) is the contact damping, and \( \sigma_2 \) is viscosity coefficient. The velocity of bristle deflection is given by

\[ \frac{dz}{dt} = v \left( 1 - \alpha(z, v) \frac{\sigma_0}{g(v)} z \cdot \text{sgn}(v) \right) \]  
(26)

in which the function \( \alpha(z, v) \) is used to capture stiction, since it just allows elastic displacement until the system reaches the break-away force. This function is described as

\[ \alpha(z, v) = \begin{cases} \alpha(z) & \text{if } v \cdot z \geq 0 \\ 0 & \text{if } v \cdot z < 0 \end{cases} \]  
(27)

with

\[ \alpha(z) = \begin{cases} 0 & \text{if } \|z\| < z_{ha} \\ \frac{1}{2} \sin \left( \pi \frac{\|z\| - z_{ba}}{z_{ba} - z_{max}} \right) + 1 & \text{if } z_{ba} < \|z\| < z_{max} \\ 1 & \text{if } z_{max} < \|z\| \end{cases} \]  
(28)

where \( z_{max} \) is the maximum bristle deflection and \( z_{ba} \) is the break-away bristle deflection. The value of \( z_{max} \) can be determined based on the steady state friction, and the relation \( z_{ba}/z_{max}=0.7 \) can be used to calculate \( z_{ba} \) [18].

### 3.5 Gonthier Model

Gonthier et al. [19] introduced a two-dimensional friction model based on LuGre approach [17]. The authors considered a force from the bending of the bristles, given by

\[ F_{br} = \sigma_0 z + \sigma_1 \frac{dz}{dt} \]  
(29)

where \( \sigma_0 \) is the stiffness and \( \sigma_1 \) is the damping coefficient. To ensure a smooth transition between the stick-slip friction regimes, an auxiliary parameter is defined as,

\[ s = e^{\left( \frac{v}{v_s} \right)^2} \]  
(30)

where \( v_s \) is the Stribecck velocity. When the bodies are sticking, the deformation rate will be equal to the relative velocity, while for sliding, the resultant friction force will approach the Coulomb friction force, \( F_C \).

\[ \frac{dz}{dt} = sv + (1-s) \left( \frac{1}{\sigma_1} F_C - \frac{\sigma_0}{\sigma_1} z \right) \]  
(31)

Coulomb friction always has the velocity direction and can be approximated by

\[ F_C = F_c \cdot \text{dir}_0(v, v_0) \]  
(32)

where \( \text{dir}_0(v, v_0) \) returns the unit vector with velocity direction, and it smooths the vector oscillations for velocities under a certain tolerance, \( v_0 \), to diminish the discontinuities in velocity direction. This tolerance velocity is considered \( v_0 = 0.01 v_s \).

\[ \text{dir}_0(v, v_0) = \begin{cases} \frac{v}{\|v\|} & \text{if } \|v\| \geq v_0 \\ \frac{v}{v_0} \left( 3 \frac{\|v\|}{2 v_0} - \left( \frac{\|v\|}{v_0} \right)^3 \right) & \text{if } \|v\| < v_0 \end{cases} \]  
(33)
This approach includes a temporal lag associated with the dwell-time dependence. To capture that phenomenon, a new state variable is defined as

\[
\dot{s}_{dw} = \begin{cases} 
\frac{1}{\tau_{dw}}(s - s_{dw}) & \text{if } s - s_{dw} \geq 0 \\
\frac{1}{\tau_{br}}(s - s_{dw}) & \text{if } s - s_{dw} < 0
\end{cases}
\]  

(34)

where \(\tau_{dw}\) is the dwell-time dynamics time constant, and \(\tau_{br} = \sigma_1 / \sigma_0\) is the bristle dynamics time constant. The time constants should be set according to the desired time delay, a large one for sticking, and a small time delay for sliding. Thus, the maximum friction force can be defined as

\[
F_{\text{max}} = F_{C} + (F_{S} - F_{C}) s_{dw}
\]

(35)

where \(F_{C}\) and \(F_{S}\) are the magnitude of Coulomb and static friction, respectively. Thus, the friction force can be expressed as

\[
F = \begin{cases} 
F_{br} + \sigma_2 v & \text{if } \|F_{br}\| \leq F_{\text{max}} \\
F_{\text{max}} \text{sgn}(F_{br}) + \sigma_2 v & \text{if } \|F_{br}\| > F_{\text{max}}
\end{cases}
\]

(36)

where \(\sigma_2\) is the viscous damping coefficient. The use of this model results in a set of ordinary differential equations that are very stiff at low relative velocities and cannot be solved using explicit ODE solvers.

### 3.6 Liang Bristle Model

This bristle friction model [20] is an extension of the model presented in [16] to the 3D space. The average deflection of the bristles is represented by a linear spring, which can stretch and rotate, and it is constrained to the tangential plane of the contact. Thus, for each individual contact, the friction force can be calculated as

\[
F = k_{b} s
\]

(37)

where \(k_{b}\) is the bristle stiffness and \(s\) is the average bristle deflection and can be expressed as

\[
s(t) = \begin{cases} 
s(t_0) + \int_{t_0}^{t} v(t) dt & \text{if } \|s\| < s_{\text{max}} \\
s_{\text{max}} \frac{v}{\|v\|} & \text{if } \|s\| \geq s_{\text{max}}
\end{cases}
\]

(38)

where \(t_0\) is the starting time of the contact, \(t\) is the current time and \(s_{\text{max}}\) is the maximum bristle deflection that can be defined as

\[
s_{\text{max}} = \begin{cases} 
\frac{s_{k_{\text{max}}}}{k_{b}} = \frac{F_{C}}{k_{b}} & \text{if } \|v\| > v_d \\
\frac{s_{s_{\text{max}}}}{k_{b}} = \frac{F_{S}}{k_{b}} & \text{if } \|v\| \leq v_d
\end{cases}
\]

(39)

where \(v_d\) is a threshold velocity, which represents the numerical boundary between the sticking and sliding regimes. Thus, there is a maximum value for bristle deflection for sticking \((s_{k_{\text{max}}})\) and another for sliding \((s_{s_{\text{max}}})\).

The major drawback of this model is related to the transition of the sticking regime to the sliding regime. This is because it corresponds a sudden decrease of the maximum value of the average bristle deflection, which can result in an abrupt change of the friction force. To stabilize the friction force, the equation (37) can be modified through the introduction of a damping term; i.e.,
where \( c_b \) is the bristle damping coefficient. When the model reaches the steady-state in the sliding mode, the friction force will be equal to Coulomb friction force. In the sticking mode, the friction force will be higher, and equal to the static friction.

### 4 COMPARISON RESULTS FOR AN ILLUSTRATIVE EXAMPLE

In this section, the classic 1-DOF spring-mass system is utilized as a numerical example of application, which allows for the analysis and comparison of the different friction models (see Figure 1). This model consists of a block with mass \( m \), which is positioned on a conveyor belt. The belt is moving with a constant velocity. The block is connected by a spring element with stiffness \( k_s \). The system dynamics is governed by the differential equation

\[
F = k_s x + m \ddot{x}
\]

The simulation parameters for the spring-mass system are presented in Table 1. The specific parameters associated with each friction model were extracted from the literature and listed in Table 2. Initially, the block is located at the origin of the \( xy \) coordinate system, and its velocity being the same as the belt.

**Table 1.** Simulation parameters for the spring-mass model.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mass of the block</td>
<td>( m )</td>
<td>1 kg</td>
</tr>
<tr>
<td>Velocity of the belt</td>
<td>( v_b )</td>
<td>0.1 m/s</td>
</tr>
<tr>
<td>Spring stiffness</td>
<td>( k_s )</td>
<td>2 N/m</td>
</tr>
</tbody>
</table>

Figures 1b to 1d show the main results produced from simulations with different friction models. In order to keep the analysis simple, the friction approaches are grouped into three classes, namely static models without stiction, static models with stiction, and dynamic models. The behavior of the system is quantified by the plots of position, relative velocity and friction force values. The results are relative to 20 s of simulation.

**Table 2.** Parameters considered for the different friction models.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Static coefficient of friction</td>
<td>( \mu_s )</td>
<td>0.15</td>
</tr>
<tr>
<td>Kinetic coefficient of friction</td>
<td>( \mu_k )</td>
<td>0.1</td>
</tr>
<tr>
<td>Coefficient of viscosity</td>
<td>( F_v )</td>
<td>0.1 Ns/m</td>
</tr>
<tr>
<td>Strubeck velocity</td>
<td>( v_s )</td>
<td>0.001 m/s</td>
</tr>
<tr>
<td>Geometry factor</td>
<td>( \delta_0 )</td>
<td>2</td>
</tr>
<tr>
<td>Tolerance velocity (Kamopp)</td>
<td>( D_v )</td>
<td>0.001 m/s</td>
</tr>
<tr>
<td>Tolerance velocity (Threlfall, Bengisu and Akay)</td>
<td>( v_0 )</td>
<td>0.001 m/s</td>
</tr>
<tr>
<td>Factor for curve shape</td>
<td>( \xi )</td>
<td>50 s/m</td>
</tr>
<tr>
<td>Tolerance velocity (Ambrósio)</td>
<td>( v_0 )</td>
<td>0.0001 m/s</td>
</tr>
<tr>
<td>Tolerance velocity (Ambrósio)</td>
<td>( v_1 )</td>
<td>0.001 m/s</td>
</tr>
<tr>
<td>Tolerance velocity (Ambrósio)</td>
<td>( v_2 )</td>
<td>0.001 m/s</td>
</tr>
</tbody>
</table>

Regarding the static models without stiction, they present a sticking phase related to the initial conditions, since the spring force magnitude is lower than the Coulomb friction force. The fact of having an exact zero relative velocity at the beginning of the simulation avoids any numerical instability for the models with a discontinuity at null velocity, as in the case of the Coulomb model and the Coulomb with viscous approach. Since these models have a constant Coulomb friction force or lower for velocities close to zero, the block exhibits a spring-like behavior with a frequency equal to \( 1/2\pi\sqrt{k_s/m} \). From the analysis of Figure 1b, it can be observed that the results corresponding to Coulomb with viscous friction shows a distinct behavior since the viscosity introduces a damping effect in the block oscillations.
For the static models with stiction, the differences are more evident. Both simulations with Coulomb with stiction and Stribeck friction present numerical instability can be observed in the friction force plot on Figure 1c. This occurs due to the changes in the velocity direction. The Coulomb model with stiction is capable of reaching the static friction only at the first peak, and because of the initial velocity, it does not stick again. Karnopp and Awrejcewicz models have similar responses, and present well-defined stick and slip phases. As the model with Stribeck friction is the only one with viscous friction component, it has faster cycles comparing with the other approaches.

Figure 1. (a) Representation of the 1-DOF spring-mass model; (b) static models without stiction; (c) static models with stiction; (d) dynamic models.
Analyzing the dynamic models, it is possible to observe that Dahl’s model is the only one that does not capture the stick-slip phenomenon. From Figure 1d, the remaining models can be divided into two groups, the first includes LuGre, Elasto-Plastic and Gonthier, and the second one includes Reset Integrator and Liang. The difference in the dynamic corresponding to these models behavior is caused by presence of a viscous component in the first set of models. Figures 2a and 2b show the plots of the friction force versus displacement and friction force versus relative velocity for the dynamic friction approaches. Comparing these models with Rabinowicz’s work [3], it can be stated that LuGre, Elasto-Plastic and Gonthier’s present the friction force as a function of displacement with a similar shape. During the pre-sliding displacement phase, the slope of the friction force is directly related to the stiffness coefficient of each model (see Figure 2a). In contrast with the static models, the dynamic approaches do not change the force direction with the change of velocity direction, as can be observed in Figure 2b. The Reset Integrator, Elasto-Plastic, and LuGre (to some extent) seem increase the friction force before sticking. In turn, the Liang’s model shows more instability in velocity, before reaching the sticking phase.

Figure 2. Dynamic models behavior: (a) Friction force versus displacement; (b) Friction force versus relative velocity

5 CONCLUDING REMARKS

A comparative study of some of the most relevant friction force models for multibody systems dynamics has been presented in this work. In the sequel of this process, the main characteristics, advantages and limitations of the static and dynamic friction force models were also analyzed. To perform a comparative study of the dynamic response of the discussed models, a 1-DOF spring-mass model was utilized as example of application.

The Coulomb friction law was the major precursor of the evolution of the friction force models. This model however shows a huge inability to capture different friction phenomena. These phenomena have a direct influence on the dynamic response of the system. To mitigate the differences between the reality and friction modelling, robust models have been evaluated. Nevertheless, in order to have more complex friction models, it is, in general, necessary to introduce larger number of parameters to fully define the physics of the friction phenomena. This study illustrates the correctly model friction in multibody systems to, analyze and to simulate them, appropriate friction models must be adopted.

REFERENCES

Spatial Impact of a Beam Attached to a Sliding Structure

Mihai Dupac

Faculty of Science and Technology
Bournemouth University
Talbot Campus, Fern Barrow, BH12 5BB, UK
mdupac@bournemouth.ac.uk

ABSTRACT
Dynamic simulation of impact in multibody system is of great interest for scientists and engineers in many different areas - robotics, biomechanics, computer animations and/or virtual reality are just a few examples. Simple or multiple impacts/collisions - contact event(s) that occurs at a common point of contact – in multibody dynamics presents many challenges. Due to the complexity of the impact dynamics, the development of various methods for predicting the behavior multibody systems (as for example kinematic chains) after collision is considered desirable. Analytical solutions of the post impact velocities (in terms of pre-impact velocities) may be obtained through the use of the classical rigid body collision theory formulated in terms of Newton’s fundamental principles (law of motion) and Coulomb’s friction model. In addition, the knowledge of some material constants such as coefficient of restitution and coefficient of friction are required. In this paper, the dynamics of a spatial impact of a rigid beam attached to a sliding structure - two-link chain - with an external surface is considered. The normal impulsive forces – considered in differential formulation of the equations of impact – are determined by combining the elastic-plastic indentation theory with the classical Hertzian contact theory. To reflect dissipation in the contact/impact area and energy loss during impact the force-deformation model includes damping. Velocity and kinetic energy are investigated for different incident impact angles of the beam in an impact process which consider friction at the contact point. The two-link chain - beam and sliding mass - impact is studied numerically. Further acquired experimental data should be compared with the numerical results for validation and generalisation.

Keywords: Impact, kinematic chain, multibody systems, dynamics, energy.

1 INTRODUCTION
The effect of impact on the dynamical behaviour of mechanical systems has been of great interest for scientists and engineers in the area of robotics. Simple or multiple impacts/contact event(s) in multibody dynamics presents many challenges [3,4], passive or dynamic walking and running are just some of various examples where an accurate modeling of impact (with the ground) represent an essential problem [18]. Some other examples involve the modeling of simultaneous constraints in kinematic chains due to their kinematic joints (involving non-linear springs, dampers, translational and rotational spring) and contact/impact of the links. Differential formulations or algebraic used to solve collision problems involves either the use the coefficient of restitution which relates the pre and post contact velocities, or the division of the collision period into a compression and restitution phase, in which case, the normal impulses are related through Poisson’s hypothesis (which works well when a tangential impact is involved). Classical approaches in constrained multibody systems for modelling impact-induced impulsive motions is based either on a smooth force distribution (impulsive force as a compliant model) or by considering an algebraic method (impulse-momentum balance equations) as shown in [12]. Due to the complexity of the impact dynamics, predicting the behavior of open kinematic chains after collision while computing post impact velocities in presence of friction, different body geometry and materials require a profound and in depth knowledge, especially when some approaches (Newton approach) may predict erroneous (energy) result as shown in
A new differential approach using a kinetic coefficient of restitution has been used [11] to address and correct the erroneous predictions. An energetically theory has been used in [15] to define a new coefficient of restitution based on the internal dissipation hypothesis of partly elastic collisions. Two dimensional impact with dry friction, tangential impact, frictional impulse (using Newton’s and Poisson’s models) and impact classification (sliding, sticking and reverse sliding) are considered in [20].

Rigid body collisions of planar kinematic chains with multiple contact points have been considered in [5,8,9]. A detailed solution procedure [9] presents the impact equations of a kinematic chain in a differential and algebraic formulation that incorporates the three definitions of the coefficient of restitution. Using the same definitions of the restitution coefficient, transverse and elastic deformations have been used in [19] to study the frictional effects for different impact angles. A kinematic chain with multiple points impacting a granular matter, with the force acting on the links penetrating the granular media described as a linear superposition of a static (resistance) force and a dynamic (frictional) force is discussed in [14]. A Lagrangian formulation for shocks and percussions created by normal reaction forces of the impacting bodies and by the forces due to the kinematic joints is presented in [4]. An algebraic formulation of a kinematic chain impacting an external surface based on an energy correction scheme [1] and the use of coefficient of restitution have been considered in [8].

In this paper, the dynamics of a rigid beam attached to a sliding structure - two-link chain - impacting an external surface is considered. Initially, the impacting system and associated coordinates and then, the normal impulsive forces – considered in differential formulation – are determined by combining the elastic-plastic indentation and Hertzian contact theory. The post and pre-impact energies and rebound velocity are investigated for different impact angles of the impacting link of the kinematic chain in an impact process which consider friction at the contact point. The obtained numerical results should/will be compared against further experimental data obtained from the experimental rig presented here.

2 IMPACTING SYSTEM

2.1 System Model - Equations of Motion with no Impact

The impacting system considered in this study is represented as a kinematic chain [5,8,9] having \( n \) interconnected equal rigid links and \( n-1 \) joints modelled as linearly-elastic torsional springs as shown in Fig. 1. \( \omega_i = u_{P_i} \cdot k \). The spring-like force [21] due to the torsional spring of stiffness \( c \) cause a joint torque \( \tau(\Delta \theta) = c \Delta \theta \), where \( \Delta \theta \) represents the joint flexion \( \theta - \theta' \) with the rest angle \( \theta \). The rigid links \( P_1, \ldots, P_n \) have the lengths \( l_{R_i} = l_{R_i} = \ldots = l_{R_i} = l \) and the torsional springs \( T_1, \ldots, T_n \) have the spring constants \( k_1, \ldots, k_n \). The end of the first link of the kinematic chain collides with a surface denoted by \( S \). For the kinematic chain shown in Fig. 1, consider \( R[i, j, k] \) a fixed inertial reference frame which gives the orientation of the impacting surface \( S \) (surfaces \( S \) can make an angle \( \theta \) with the horizontal) and \( R[i, j, i, j'] \), \( k \) a set of reference frames attached to the links \( P_i \), where \( i, j, k \) and \( i, j', k \) are the unit normal vectors attached to the frames \( R \) and \( R \) respectively for \( i = 1, 2, \ldots, n - 1 \). The orientation of the link \( i \) with respect to the link \( i+1 \) is given by the angle \( \theta_{P_{i+1}} \) between the vectors \( \hat{i} \) and \( \hat{i+1} \). The coordinates of the mass centre \( C_i \) of the link \( P_i \), are denoted by \( x_{C_i} \) and \( y_{C_i} \), while the orientation/angles of the link \( P_{i+1} \) with respect to its adjacent link \( P_i \) are denoted by \( \theta = \theta_{P_{i+1}} \), \( i = 1, 2, \ldots, n \). The impacting link of the kinematic chain has three degrees of freedom, that is, two translations about \( Ox \) and \( Oy \) denoted by \( d_{R_i} \) and \( d_{R_i} \), a rotational one denoted by \( \phi_{R_i} \). The
impacting link will rebound as a result of the collision, leading to different outcomes depending on the initial/boundary conditions, contact force, and friction coefficient at contact point.

![Figure 1. General representation of an impacting kinematic chain with joints modelled as linearly-elastic torsional springs](image)

One can write the generalised coordinates of the impacting system in a vector format by

$$\mathbf{q} = \{\theta_{SP_1}, \theta_{P_1P_2}, ..., \theta_{P_{n-1}P_n}, d_{P_1}, d_{P_n}\}^T \quad (1)$$

The position of each rigid link $P_1, ..., P_n$ can be expressed with respect to the generalised coordinates in Eq.(1). Writing the generalised speed of the rigid links $P_1, ..., P_n$ in a vector format as

$$\mathbf{u} = \dot{\mathbf{q}} = \{\dot{u}_{SP_1}, u_{P_1P_2}, ..., u_{P_{n-1}P_n}, u_{P_1}, u_{P_n}\}^T \quad (2)$$

where $u_{P_i} = \dot{d}_{P_i}, u_{P_i} = \dot{d}_{P_i}$, one can calculate the angular velocity of each rigid link $P_i, i=1,2,...,n-1$ with respect to the fixed inertial reference frame by $\mathbf{\omega}_i = u_{P_i} \mathbf{k}$. The normal and tangential components of the impulse vector at contact can be obtained by integrating the normal contact force $F_n$ and the tangential contact force $F_t$ as

$$\mathbf{\tau} = \begin{bmatrix} \int_0^t F_n dt \\ \int_0^t F_t dt \end{bmatrix} \quad (4)$$

The velocity of the end of the first link of the kinematic chain collides with the surface S can be expressed in a vector format using

$$\mathbf{v} = v_n \mathbf{i} + v_t \mathbf{j} \quad (5)$$

where $v_n$ is the normal velocity and $v_t$ is the tangential velocity at contact/impact. The gravitational forces acting on the links $i=1,2,...,n-1$ are $\mathbf{G}_i = -m_i g \mathbf{j}$, and the Lagrangian of the system is defined as $L=T-V$ where $T$ is the total energy and $V$ is the potential energy.

One can write the Lagrange differential equation of motion with no impact for the kinematic chain as
\[
\frac{d}{dt} \left( \frac{dL}{dq} \right) - \frac{dL}{dq} = 0
\]  \hspace{1cm} (6)

### 2.2 Impact Force Computation - Equations of Motion with no Impact

Considering the classical Hertzian contact theory as presented in [2,6] and the elastic-plastic indentation theory discussed in [5,13] one can determine the normal impulsive force \( F_n \) of the kinematic chain at impact point. The normal contact force \( F_n \) can be expressed as a linear relation between a critical value of the impact force \( F_c \) and the plastic deformation \( q_p \) at contact by

\[
F_n = F_c + 2\pi RHq_p
\]

where \( R \) is the radius of the end impacting link and \( H \) represents the plastic property of the material approximated with the Brinell hardness. The term \( F_c \) in Eq.(1) be calculated in terms of the yield stress \( \sigma_y \) by

\[
F_c = \left(2\pi R\right)^{\frac{3}{2}} \frac{\sigma_y^3}{k^2}
\]

where \( k = \frac{2\sqrt{R}}{3} \frac{E}{1-\nu^2} \) is expressed in terms of radius \( R \) of the end impacting link, Poisson’s ratio \( \nu \) and the Young’s module \( E \). The normal contact force \( F_n \) can be expressed using Hertz’s low in terms of the elastic deformation \( q_e \) by

\[
F_n = k\sqrt{q_e}
\]

The total normal deformation \( q_e \) of the elasto-plasto impact can be expressed as the sum of elastic and plastic deformation by \( q_n = q_p + q_e \). Since the critical deformation \( q_c \) corresponds to a critical force \( F_c \), a maximum deformation appears for a maximum applied force \( F_n \). The equations of motion with impact for the kinematic chain in Fig.1 can be expressed as in [5,9,12], that is

\[
M(q)\ddot{q} + C(q, \dot{q}) + G(q) = \begin{bmatrix} T \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} D_1(q) \\ 0 \\ D_2(\theta) \end{bmatrix} F
\]

where \( q, \dot{q} \) and \( \ddot{q} \) are the vectors of generalized coordinates, velocities and accelerations respectively, \( M \) is the mass matrix, \( G \) vector of gravity terms, \( T \) is the joint moments vector, and \( F \) is the vector of contact forces. Using Newton–Euler equations for a specific kinematic chain, all the matrices \( M, G, T, D_1, D_2, \) and \( F \) can be calculated [8,9,14]. Considering \( F = F_n1 + F_t j \) the impact force at the impact point, the possible cases of motion at the end point of the impacting chain link [5,9,12] can be described by:

I. the end is slipping along the impacting surface (a nonzero tangential velocity at collision), in which case a dry friction coefficient can be used to model the link impact [20]. The tangential component of the impact force can be expressed as \( F_t = -\mu \text{Sign}(v_t)F_n \)

II. the end is not slipping along the impacting surface, that is, tangential velocity \( v_t = a_t = 0 \) subject to \( |F_t/F_n| \leq \mu \)
3 APPLICATION AND RESULTS

In this section the rigid body collision with friction of 3D kinematic chain - rigid beam attached to a sliding structure (Fig. 2) - having the joint(s) modelled as linearly-elastic torsional springs is considered. The torsional spring has a stiffness of \(98 \text{ N} \cdot \text{m/ rad}\). The Cartesian reference frame \(R \{i, j, k\}\) (denoted also by \(xOy\)) shown in Fig. 2a is is attached to the surface \(S\) while the two mobile reference frames denoted by \(R_1 \{i_1, j_1, k_1\}\) and \(R_2 \{i_2, j_2, k_2\}\) are attached to the links 1 (rigid beam) and link 2 (sliding mass) at the end of the link and at the lower part of the sliding mass respectively. The vectors \(k, k_1,\) and \(k_2\) are perpendicular to the plane of view and therefore cannot be visualized (observed) in the Fig. 2.

The contact surface \(S\) has a variable orientation \(\phi\) with the horizontal. The chain include one slender member, that is, an aluminium bar with a density of \(2720 \text{ kg/m}^3\), diameter of \(0.025 \text{ m}\) and length of \(0.3 \text{ m}\), connected with a rotational/torsional spring (located at \(0.015 \text{ m}\) from the link end) to the sliding structure/mass. The mass of the link 1 is \(m_1 = 0.041 \text{ kg}\), while the mass of the sliding structure can be adjusted (the sliding carriage may have additional weights added as shown in Fig. 2a). The coefficient of static friction is \(\mu_s = 1.17\) and the coefficient of dynamic/kinetic friction \(\mu_k = 1.40\) (for dry aluminium).

![Figure 2. Impacting kinematic chain - rigid beam attached to a sliding structure/mass - with joints modelled as linearly-elastic torsional springs](image)

The velocity of the impacting end of the aluminium bar (impacting point on link 1) on the \(Ox\) and \(Oy\) axes is denoted by \(v_x\) and \(v_y\) respectively. The total kinetic energy for the impacting system is the sum between the kinetic energy \(T_1\) of the link 1 and the kinetic energy \(T_2\) of the sliding structure. A gravitational acceleration of \(9.807 \text{ m/s}^2\) is considered for the simulation. For each simulation, the impacting system (beam and sliding mass) is released such as the distance between impacting end of the beam and the surface \(S\) is constant and equal with \(d = 200 \text{ mm}\). When released, the impacting beam has no initial angular velocity, that is, \(\omega_i = 0 \text{ rad/sec}\). An constant impact angle of \(\theta_{Spl} = 0.39 \text{ rad}\) between the impacting beam and the vertical direction at the impact point has been considered. Two different values for the orientation \(\phi\) of the contact surface \(S\) with the horizontal have been considered for each simulation. The orientation angles with horizontal direction of the surface \(S\) are \(\theta_S = 0 \text{ rad}\) and \(\theta_S = 0.1 \text{ rad}\) respectively. The For the simulation results shown in Fig. 3 the symbols “*” and “■” have been used to shown the values obtained immediately before and immediately after the impact.
Figure 3. Impacting kinematic chain - rigid beam attached to a sliding structure/mass - with joints modelled as linearly-elastic torsional springs

The $Ox$ and $Oy$ velocities vs. the angle of impact, angular velocities vs. the angle of impact, and kinetic energy vs. the angle of impact, computed for post and pre-impact are shown in Fig.4.

4 CONCLUSIONS

In this paper rigid body collision with friction of 3D kinematic chain - having the joints modelled as linearly-elastic torsional springs - is considered. Numerical results – including kinetic energy $T$ and angular post and pre-impact velocities - of the collision of an impacting system, i.e., beam and sliding mass with a flat surface having a variable orientation $\varphi$ with the horizontal, in the presence of gravity and considering friction at the contact point are presented. The obtained numerical results will be compared against further experimental data - as detailed below – in order to generalise the obtained results from the simulation.

5 FUTURE WORK

To validate and generalise the numerical results experimental data should be obtained and compared with the simulations. The rig shown in Fig. 4a will be used for data acquisition. The rig is basically a vertical sliding motion mechanism/frame having a built in handle designed to position the frame in the desired released position, able to carry various masses and on the lower edge the impact beam. Smooth sliding is ensured due to the use of slide rails with ball bearings on each side.
Acquisition of impact data including pre and post impact velocities will be/are obtained from the data acquisition system shown in Fig. 4b, that is, a linear resistive transducer responsible for measuring the travel distance of the sliding mass and impact beam, a 2-Axis force platform PS-2142, a PASSPORT interface and Capstone data acquisition software for recording/measuring the impact response (in both normal direction and tangential direction) with a resolution of 0.1 N for the beam and sliding mass mechanical system.

REFERENCES


Synchronization-based state observer for impacting multibody systems using switched geometric unilateral constraints

Michael Baumann†, Remco I. Leine∗

† Institute of Mechanical Systems
ETH Zurich
8092 Zurich, Switzerland
baumann@imes.mavt.ethz.ch

∗ Institute for Nonlinear Mechanics
University of Stuttgart
70569 Stuttgart, Germany
remco.leine@inm.uni-stuttgart.de

ABSTRACT
We present a new design of a state observer for linear time-invariant multibody systems subjected to unilateral constraints using only the information of the impact time. It extends the approach presented in [4] with the new concept of switched geometric unilateral constraints. These constraints introduce constraint forces in the kinematic equation which render the generalized coordinates discontinuous. The introduction of position jumps improves the synchronization rate and expands the applicability of the observer.

A master–slave synchronization setup is used for which the unidirectional coupling between the master (observed system) and the slave system (observer) consists only of the impact time information. The dynamics of the slave system is shown to be attractively incrementally stable due to the switched constraints. The observer replicates the full state for all initial conditions, also in the presence of accumulations points (Zeno behavior) and in the vicinity of grazing impacts. The results are illustrated using two examples of impacting mechanical systems.

Keywords: observer design, master–slave synchronization, switched unilateral constraint, measure differential inclusion, non-smooth dynamics.

1 Introduction
In this paper we introduce a new type of constraints called switched geometric unilateral constraints, which allow for position jumps. Sufficient conditions are shown for which linear time-invariant systems subjected to switched geometric unilateral constraints are attractively incrementally stable. This result is used to design a state observer based on master–slave synchronization which uses only the Boolean information of the impact time instants. The proposed observer improves the observer presented in [4] by extending the applicability and increasing the synchronization rate.

The multibody systems considered in this paper are subjected to unilateral constraints. The states of these non-smooth systems are discontinuous and assumed to be special functions of locally bounded variation [2]. The dynamics can conveniently be described using the framework of measure differential inclusions [15, 14, 5, 1, 9] and phenomena such as accumulation points (Zeno behavior) do not have to be excluded.

The new concept of switched geometric unilateral constraints are accompanied by constraint forces which act on the kinematic equation. These constraints are used to improve the observer presented in [4] by extending the applicability and increasing the synchronization rate. Using these new type of constraints, the generalized coordinates become discontinuous, which raises the question of a suitable metric for the position jumps. The same problem arises when considering the Gear-Gupta-Leimkuhler method to enforce the geometric constraints when simulating mechanical system with impacts [8, 10, 19]. The GGL approach classically projects the generalized coordinates to the non-penetration constraint using the identity metric or the metric induced by the mass matrix, which is
non-energy consistent. Energy considerations imply the use of the stiffness matrix for the metric, which is only applicable if the stiffness matrix is known and positive definite.

The observer uses a master–slave synchronization setup and the decay of the synchronization error is based on the property of attractive incremental stability of the slave system. Incremental stability is a system property and several similar notions have been presented in the literature [21, 6, 7, 3]. Incremental stability is beneficial in many control problems such as stabilization, output regulation problems, synchronization and observer design [16, 20, 17, 11].

The proposed observer uses only the Boolean impact time information and no additional continuous measurement is necessary. Mechanical systems without any feedback are generally not strictly passive. More precisely, the transfer matrix of the linear part of the system is positive real, but not strictly positive real. Therefore, invariance like results are necessary for this class of mechanical systems to show the property of incremental stability.

This paper is organized as follows. The dynamics of the observed system is described in Section 2 using the framework of measure differential inclusions. The main result is presented in Section 3, where the design of the state observer is shown. Sufficient conditions are provided for which the dynamics of the observer is attractively incrementally stable. The results are illustrated with simulations of two examples of impact oscillators in Section 4 and final conclusions are given in Section 5.

2 Dynamics of the observed system

In this section we describe the dynamics of the system for which we will design a state observer. The dynamics is expressed in the form of a measure differential inclusion which allows us to describe the impulsive and non-impulsive dynamics in a combined form [15]. We consider an $n$-DOF linear time-invariant multibody system subjected to geometric unilateral constraints. The generalized coordinates $q(t)$ are absolutely continuous in time and the generalized velocities $u(t)$ are discontinuous due to the unilateral constraints and are assumed to be special functions of bounded variation [2]. The dynamics of the observed system is given by

$$
\begin{align*}
dq &= u \, dt \\
M \, du &= (-Cu - Kq + f(t)) \, dt + W\lambda \, dt + W\Lambda \, d\eta.
\end{align*}
$$

The differential measure $dq = \dot{q} \, dt$ has only a density with respect to the Lebesgue measure $dt$, whereas the differential measure $du = \dot{u} \, dt + (u^+ - u^-) \, d\eta$ has additionally a density with respect to an atomic measure $d\eta$. The atoms correspond to the impact times and the atomic measure can be interpreted as the sum of Dirac point measures. See [5, 14, 1] for a detailed description of the framework of measure differential inclusions for mechanical systems.

The mass matrix $M = M^T$ and the stiffness matrix $K = K^T$ are assumed to be symmetric. The system matrices $M, C, K \in \mathbb{R}^{n \times n}$ are positive definite and time invariant. The external forcing $f(t)$ is independent of the state. The mechanical system (1) is subjected to $m$ geometric unilateral constraints. The generalized coordinates $q(t)$ are absolutely continuous in time and the generalized velocities $u(t)$ are discontinuous due to the unilateral constraints and are assumed to be special functions of bounded variation [2]. The dynamics of the observed system is given by

$$
\begin{align*}
dq &= u \, dt \\
M \, du &= (-Cu - Kq + f(t)) \, dt + W\lambda \, dt + W\Lambda \, d\eta.
\end{align*}
$$

The differential measure $dq = \dot{q} \, dt$ has only a density with respect to the Lebesgue measure $dt$, whereas the differential measure $du = \dot{u} \, dt + (u^+ - u^-) \, d\eta$ has additionally a density with respect to an atomic measure $d\eta$. The atoms correspond to the impact times and the atomic measure can be interpreted as the sum of Dirac point measures. See [5, 14, 1] for a detailed description of the framework of measure differential inclusions for mechanical systems.

The mass matrix $M = M^T$ and the stiffness matrix $K = K^T$ are assumed to be symmetric. The system matrices $M, C, K \in \mathbb{R}^{n \times n}$ are positive definite and time invariant. The external forcing $f(t)$ is independent of the state. The mechanical system (1) is subjected to $m$ geometric unilateral constraints. The generalized force directions $w_i$ are collected in the matrix $W = \{w_1, w_2, \ldots, w_m\} \in \mathbb{R}^{n \times m}$ and are assumed to be linearly independent and time invariant. The constraint forces $\lambda$ and constraint impulses $\Lambda$ are given by constitutive laws defined using the local kinematic quantities

$$
\begin{align*}
\text{constraint distance:} & \quad g(q, t) = W^T q + \nu(t), \\
\text{constraint velocity:} & \quad \gamma(u, t) = W^T u + \dot{\nu}(t),
\end{align*}
$$

where $\nu(t)$ is an absolutely continuous function in time.
The \( i \)-th geometric unilateral constraint restricts the sign of the constraint distance \( g_i \geq 0 \). Its force law \( 0 \leq \lambda_i \perp g_i \geq 0 \), also referred to as Signorini’s law, can be written on velocity level (see [9]) as

\[
-\lambda_i \in \begin{cases} 
\partial \Psi_{R^+_0}(\gamma_i) & \text{if } g_i(q, t) = 0, \\
0 & \text{if } g_i(q, t) > 0, 
\end{cases}
\]

(4)

where \( \partial \Psi_{R^+_0} \) is the subdifferential of the indicator function \( \Psi_{R^+_0} \) on the set \( R^+_0 \). Therefore, the admissible set of system (1) is given by \( A = \{ (q, u) \in \mathbb{R}^{2n} \mid g(q) \geq 0 \} \). The impact law for the impulsive unilateral constraint forces \( \Lambda \) is given by the inclusion

\[-\Lambda \in \mathcal{H}_g(\bar{\gamma}).\]

(5)

The set-valued map \( \mathcal{H}_g \) puts a relationship between the dual variables \( \bar{\gamma} := \frac{1}{2}(\gamma^+ + \gamma^-) \) and \( \Lambda \) (see [13]). The index indicates the dependence on the set of closed contacts (given by \( g \)). It will become apparent in Subsection 3.2 that monotonicity of the impact map is a useful property of the impact map. The definition of monotone maps can be found in [18].

**Definition 1.** The set-valued map \( \mathcal{H}_g(\bar{\gamma}) \) is called *monotone* if it fulfills

\[
(\Lambda_1 - \Lambda_2)^T(\bar{\gamma}_1 - \bar{\gamma}_2) \leq 0 \quad \forall -\Lambda_1 \in \mathcal{H}_g(\bar{\gamma}_1), \forall -\Lambda_2 \in \mathcal{H}_g(\bar{\gamma}_2).
\]

(6)

Commonly used impact laws such as the generalized Newton’s impact law [13] or the generalized Poisson’s impact law [4] are monotone under some mild assumptions. The monotonicity property will directly be used in Subsection 3.2 to show the decrease of the velocity error between the observer and the observed system.

### 3 Observer design

We present the design of a state observer for the class of mechanical systems described in Section 2. We assume that the model of the observed system is known, and the only available measurement is the time information when the impacts occur. We use a master–slave synchronization setup, where the observed system is the master system and state of the slave system (observer) is the estimate.

In the first step, the dynamics of the slave system is presented. Secondly, it shown that the master and the slave system have the same solution if they are initialized with the same initial conditions. Finally, we show that all solutions of the slave system converge to each other and therefore also to the solution of the master system.

#### 3.1 Observer dynamics

The model of the observer (slave system) is a replica of the observed system (master system) except for the constraints. The observer system is subjected to *switched geometric unilateral constraints* and the dynamics is described by

\[
\begin{pmatrix} K & 0 \\ 0 & M \end{pmatrix} \begin{pmatrix} dq \\ du \end{pmatrix} = \begin{pmatrix} Ku \\ -Cu - Kq + f(t) \end{pmatrix} dt + \begin{pmatrix} W\sigma \\ W\lambda \end{pmatrix} dt + \begin{pmatrix} W\Theta \\ W\Lambda \end{pmatrix} d\eta.
\]

(7)

The new concept of switched geometric unilateral constraints extends switched kinematic unilateral constraints (introduced in [4]) with position jumps by introducing the constraint forces \( \sigma \) and impulses \( \Theta \) in the kinematic equation. Therefore, the generalized velocities \( u(t) \) and constraint distances \( \gamma(u(t), t) \) are generally no longer the time derivative of the generalized coordinates \( q(t) \) and constraint distances \( g(q(t), t) \), respectively.
Furthermore, the generalized coordinates $q(t)$ are no longer absolutely continuous, but they are assumed to be functions of special locally bounded variation.

The switched kinematic unilateral constraints with the constraint forces $\lambda$ and impulses $\Lambda$ are kinematic unilateral constraints, which are switched on and off by an external Boolean switching function $\chi(t) : \mathbb{R} \rightarrow \{0, 1\}$. The $i$-th constraint imposes a kinematic unilateral constraint $\gamma_i \geq 0$ whenever the corresponding external Boolean switching function $\chi_i(t) = 1$. Its force law is described by the inequality complementarity

$$-\lambda_i \in \begin{cases} \partial \Psi_{\mathbb{R}_0^+}(\gamma_i) & \text{if } \chi_i(t) = 1, \\ 0 & \text{if } \chi_i(t) = 0. \end{cases}$$

The impact law for the constraint impulses $\Lambda$ is given by the inclusion (see [13])

$$-\Lambda \in H_{\chi(t)}(\dot{\gamma}).$$

The operator $H_{\chi(t)}$ is chosen such that it is identical to the operator $H_g$ in (5) if the same contacts are closed or switched on, respectively.

The $i$-th switched geometric bilateral constraint ensures that the $i$-th constraint is closed when $\chi_i(t) = 1$ and imposes no constraint otherwise. Therefore, the constitutive laws for the constraint forces $\sigma$ and the constraint impulses $\Sigma$ of the switched geometric bilateral constraint are given by

$$-\sigma_i \in \begin{cases} \partial \Psi \{g_i\} & \text{if } \chi_i(t) = 1, \\ 0 & \text{if } \chi_i(t) = 0, \end{cases}$$

$$-\Sigma_i \in \begin{cases} \partial \Psi \{g_i^+\} & \text{if } \chi_i(t) = 1, \\ 0 & \text{if } \chi_i(t) = 0. \end{cases}$$

The switched geometric unilateral constraints are ‘time-triggered’ and $\chi(t)$ is an external input independent of the constraint distances. The ‘state-triggered’ geometric unilateral constraints of the observed system, however, are closed if the corresponding constraint distances vanish. In order to relate these two types of constraints, we make the following definition.

**Definition 2.** The Boolean switching functions $\chi(t)$ are called to be *generated* by $g(t)$ if every component $\chi_i(t)$ of $\chi(t)$ fulfills

$$\chi_i(t) = \begin{cases} 1 & \text{if } g_i(t) = 0, \\ 0 & \text{if } g_i(t) > 0. \end{cases}$$

If the observer is initialized with the same initial conditions as the observed system and the switching functions $\chi(t)$ are generated by the constraint distances of the observed system, then both solutions are identical. This statement is the content of the following proposition.

**Proposition 1.** Let the master system be described by (1)–(5) and let the slave system be described by (7)–(11) together with (2)–(3). Let \( \left( \begin{array}{c} q_m(t) \\ u_m(t) \end{array} \right) \) be the solution of the master system for the initial conditions $\left( \begin{array}{c} q_{m}(t_0) \\ u_{m}(t_0) \end{array} \right) = \left( \begin{array}{c} q_0 \\ u_0 \end{array} \right) \in \mathcal{A}$ and let $\left( \begin{array}{c} q_s(t) \\ u_s(t) \end{array} \right)$ be the solution of the slave system for the same initial conditions $\left( \begin{array}{c} q_s(t_0) \\ u_s(t_0) \end{array} \right) = \left( \begin{array}{c} q_0 \\ u_0 \end{array} \right)$. Then, both solutions are identical if the Boolean switching functions $\chi(t)$ in (8)–(11) are generated by the constraint distances $g_m(t)$ of the master system.
Proof. The only difference between the master and the slave system are the constraints. The condition that \( \chi(t) \) is generated by \( g_m \) directly implies that the force laws (4) and (8) as well as the impact laws (5) and (9) are identical. For the constraint forces \( \sigma \) and constraint impulses \( \Sigma \) we seek an explicit expression in order to show that both vanish for the considered solutions.

We define \( I \) as the index set of closed constraints at a certain point in time as \( I(t) = \{ i \mid \chi_i(t) = 0 \} \) and introduce the following notation. A subscript \( I \) indicates that only the closed constraints are considered, e.g. \( W_I = \{ \ldots, w_i, \ldots \} \), \( g_{s,I} = (\ldots, g_{s,i}, \ldots)^T \), where \( i \in I \).

We multiply the kinematic equation in (7) from the left by \( W_I^T K^{-1} \) and obtain

\[
W_I^T \left( \dot{q}_s \ dt + (q_{s,i}^+ - q_{s,i}^-) \ d\eta \right) = W_I^T u_s \ dt + W_I^T K^{-1} W \sigma \ dt + W_I^T K^{-1} W \Sigma \ d\eta.
\]

Substituting (2)–(3) and using the absolute continuity of \( \nu(t) \) yields

\[
\dot{g}_{s,I} \ dt + (g_{s,i}^+ - g_{s,i}^-) \ d\eta = \gamma_{s,I} \ dt + W_I^T K^{-1} W \sigma \ dt + W_I^T K^{-1} W \Sigma \ d\eta.
\]

The force law (10) (written on velocity level) and the impact law (11) imply \( \dot{g}_{s,I} = 0 \) and \( g_{s,i}^+ = 0 \), since the inverse of \( \partial \Psi (0) \) is the zero function, and we obtain \( \sigma_I = -L_I^{-1} \gamma_{s,I} \) and \( \Sigma_I = -L_I^{-1} g_{s,I} \), where \( L_I := W_I^T K^{-1} W_I \). We define the matrix \( P_I \) as the matrix obtained by taking the identity matrix \( P \in \mathbb{R}^{m \times m} \) and removing all columns with an index \( i \notin I \). The force and impact laws state \( \sigma_i = \Sigma_i = 0 \ \forall i \notin I \), which implies \( \sigma = P_I \sigma_I \) and \( \Sigma = P_I \Sigma_I \). Together with \( \gamma_{s,I} = P_I^T \gamma_s \) and \( g_{s,I} = P_I^T g_s \), an explicit form for \( \sigma \) and \( \Sigma \) is obtained as

\[
\begin{align*}
\sigma &= -P_I L_I^{-1} P_I^T \gamma_s, \\
\Sigma &= -P_I L_I^{-1} P_I^T g_s,
\end{align*}
\]

where the product of matrices \( P_I L_I^{-1} P_I^T \in \mathbb{R}^{m \times m} \) is positive semi-definite.

The absolute continuity of the generalized coordinates \( q_m \) of the master system, and therefore of the generalized constraint distances \( g_m \), implies \( g_{m,I} = 0 \) for a given set \( I \). Together with (14), we obtain that \( q_m^+ = q_m^- \) implies \( \Sigma = 0 \), which yields \( q_m^+ = q_m^- \) almost everywhere w.r.t. \( d\eta \). Analogously, the absolute continuity of \( q_m \) implies \( \gamma_{m,I} = 0 \) for a given set \( I \). Together with (13), we obtain that \( u_m = u_s \) implies \( \sigma = 0 \), which yields \( \dot{q}_m = \dot{q}_s \) almost everywhere w.r.t. \( dt \). Finally, we obtain \( \frac{dq_m}{du_m} = \frac{dq_s}{du_s} \) together with \( (q_m(t_0), u_m(t_0)) = (q_s(t_0), u_s(t_0)) \), which concludes the proof. \( \square \)

In order that the observer (slave) provides an estimate of the observed system (master), the systems need to synchronize with each other. The master-slave synchronization is based on the attractive incremental stability of the observer, which is presented in the following subsection.

### 3.2 Attractive incremental stability

Attractive incremental stability (a.i.s.) is a stability property of dynamical systems which implies that all solution curves are globally uniformly attractively stable. Therefore, all solution curves approach each other and remain close in the sense of Lyapunov for all initial conditions and the information of the initial condition is lost. Here, we consider the definition of a.i.s. for measure differential inclusions as presented in [4]. Other notions of incremental stability have been presented in literature, see e.g. [3, 22].

In order to state the a.i.s. of the observer, we will make the following assumptions:
A1 The switching functions $\chi(t)$ are generated by an absolutely continuous function.

A2 The external forcing $f(t)$ is bounded, i.e. $\sup_{t \in \mathbb{R}} \|f(t)\| \leq f_{\text{max}}$ for a given bound $f_{\text{max}} < \infty$.

A3 The impact map $\mathcal{H}_\chi(t)(\dot{\gamma})$ is monotone for any $\chi(t)$.

Assumption A1 guarantees that all intervals during which a constraint is switched off are open time intervals. This property is used for the existence of solutions, but this is not in the scope of this paper. The following theorem states the a.i.s. of the slave system under the previous assumptions and is an extension of the corresponding theorem in [4].

**Theorem 1.** System (7)–(11) together with (2)–(3) for given switching functions $\chi(t)$ is attractively incrementally stable if the Assumptions A1–A3 are fulfilled.

**Proof.** We will proof the property of a.i.s. by showing that all solution curves are globally uniformly attractively stable. Therefore, consider two arbitrary solutions $(q_1(t), u_1(t))$ and $(q_2(t), u_2(t))$ for given switching functions $\chi(t)$. The position and velocity errors are given by $e = q_1 - q_2$ and $v = u_1 - u_2$, where the index is used to distinguish between the two solutions. The error dynamics follows from (7) as

$$
\begin{pmatrix}
K & 0 \\
0 & M
\end{pmatrix}
\begin{pmatrix}
de \\
dv
\end{pmatrix} =
\begin{pmatrix}
Kv \\
-Cv - Ke
\end{pmatrix} dt +
\begin{pmatrix}
W(\sigma_1 - \sigma_2^T) \\
W(\lambda_1 - \lambda_2)
\end{pmatrix} dt +
\begin{pmatrix}
W(\Sigma_1 - \Sigma_2) \\
W(\Lambda_1 - \Lambda_2)
\end{pmatrix} d\eta.
$$

(15)

The constraints forces and impulses are given by (8)–(11), where the switching functions $\chi(t)$ are the same for both considered solutions. We introduce the Lyapunov function

$$
V(e, v) = \frac{1}{2} \begin{pmatrix} e & v \end{pmatrix}^T \begin{pmatrix} K & 0 \\
0 & M
\end{pmatrix} \begin{pmatrix} e \\
v
\end{pmatrix} = \|v\|^2_M + \frac{1}{2} \|e\|^2_K,
$$

(16)

which is positive definite on the error states gives a notion of distance between these two solutions. We evaluate $V$ along solutions $e(t), v(t)$ and the differential measure $dV$ contains the densities with respect to the measures $dt$ and $d\eta$. Using the symmetry of $K$ and $M$, the differential measure $dV$ can be written as $dV = \begin{pmatrix} e & v \end{pmatrix}^T \begin{pmatrix} K & 0 \\
0 & M
\end{pmatrix} \begin{pmatrix} de \\
dv
\end{pmatrix}$, where $\dot{e} := \frac{1}{2} (e^+ + e^-)$ and $\dot{v} := \frac{1}{2} (v^+ + v^-)$. Substituting the error dynamics (15) and using the local kinematic quantities (2)–(3), we obtain

$$
dV = \begin{pmatrix} e & v \end{pmatrix}^T \begin{pmatrix} K & 0 \\
0 & M
\end{pmatrix} \begin{pmatrix} de \\
dv
\end{pmatrix} = \|v\|^2_C dt + \begin{pmatrix} g_1 - g_2 \\
\gamma_1 - \gamma_2
\end{pmatrix}^T \begin{pmatrix} \sigma_1 - \sigma_2 \\
\lambda_1 - \lambda_2
\end{pmatrix} dt + \begin{pmatrix} g_1 - g_2 \\
\gamma_1 - \gamma_2
\end{pmatrix}^T \begin{pmatrix} \Sigma_1 - \Sigma_2 \\
\Lambda_1 - \Lambda_2
\end{pmatrix} d\eta,
$$

where $\bar{g} := \frac{1}{2} (g^+ + g^-)$ and $C := \frac{1}{2} (C + C^T)$ is the symmetric part of the damping matrix $C$. The terms $(g_1 - g_2)^T (\sigma_1 - \sigma_2)$ and $(\gamma_1 - \gamma_2)^T (\lambda_1 - \lambda_2)$ vanish, since the force laws (8) and (10) are given by complementarity conditions. Similarly, the impact law (11) yields $(g_1 - g_2)^T (\Sigma_1 - \Sigma_2) = \left(\frac{1}{2} g_1 - \frac{1}{2} g_2\right)^T (\Sigma_1 - \Sigma_2)$. Therefore, the differential measure $dV$ simplifies to

$$
dV = -\|v\|^2_C dt + \left(\frac{1}{2} g_1 - \frac{1}{2} g_2\right)^T (\Sigma_1 - \Sigma_2) d\eta + (\bar{\gamma}_1 - \bar{\gamma}_2)^T (\Lambda_1 - \Lambda_2) d\eta.
$$
The constraint impulses \( \Sigma_1 \) and \( \Sigma_2 \) are explicitly derived in the proof of Proposition 1. Furthermore, the impact map \( H_{x(t)}(\gamma) \) in (9) is monotone according to Assumption A3, which implies \((\gamma_1 - \gamma_2)^T(\Lambda_1 - \Lambda_2) \leq 0\) according to (6), and we obtain

\[
dV \leq -\|v\|_C^2 dt - \frac{1}{2} g_1^2 - g_2^2 \|P_l L_i^{-1} p_i^T\|_d\eta.
\]

The matrix \( \bar{C} \) is positive definite and \( P_l L_i^{-1} P_i^T \) is positive semi-definite. Hence, we have \( dV \leq 0 \) and the equilibrium at the origin is uniformly stable [14]. Furthermore, since the Lyapunov function \( V \) is bounded from below and non-increasing, the limit

\[
V_\infty := \lim_{t \to \infty} V(e(t), v(t)) = \lim_{t \to \infty} \left( \frac{1}{2} \|v(t)\|_M^2 + \frac{1}{2} \|e(t)\|_K^2 \right)
\]

exists and lies in the interval \( 0 \leq V_\infty \leq V(e^-(t_0), v^-(t_0)) \).

The Lyapunov function \( V(e(t), v(t)) \) tends to an absolutely continuous function (constant function) and both summands in (16) have a non-positive density with respect to \( d\eta \).

Therefore, we conclude that \( \|v(t)\|_M^2, \|e(t)\|_K^2 \) and, hence, \( v(t) \) and \( e(t) \) are asymptotically absolutely continuous as well. From (17) and (18) follows

\[
V_\infty - V(e^-(t_0), v^-(t_0)) \leq - \lim_{t \to \infty} \int_{[t_0, t]} \|v\|_C^2 dt - \lim_{t \to \infty} \int_{[t_0, t]} \frac{1}{2} |g_1^2 - g_2^2\|_{L_i^{-1}} \ d\eta.
\]

Since the left-hand side in (19) is finite, we deduce that

\[
\lim_{t \to \infty} \int_{[t_0, t]} \|v\|_C^2 dt < \infty.
\]

We cannot invoke Barbalat’s lemma [12] since the solution \( v(t) \) is not uniformly continuous in time. The extension of this lemma is presented in [4] for the class of asymptotically absolutely continuous functions to which \( v(t) \) belongs. Applying the extended Barbalat’s lemma to (20) yields \( \lim_{t \to \infty} \|v(t)\|_C^2 = 0 \). The positive definiteness of \( C \) implies \( \lim_{t \to \infty} v(t) = 0 \). Substituting the limit of \( v \) into (18) yields \( \lim_{t \to \infty} e(t) = c \) for some \( c \) satisfying \( V(e^-(t_0), v^-(t_0)) \geq \frac{1}{2} \|c\|_K^2 \geq V_\infty \geq 0 \). In the next step, we show that \( c = 0 \).

Therefore, we integrate the equality of measures (15) over a time interval \( \Delta t = [t, t + \Delta t] \) with an arbitrary \( \Delta t > 0 \) and obtain

\[
\begin{pmatrix}
K & 0 \\
0 & M
\end{pmatrix}
\begin{pmatrix}
e^+(t + \Delta t) - e^-(t) \\
v^+(t + \Delta t) - v^-(t)
\end{pmatrix} = \int_{\Delta t} \left( -Cv - Ke \right) dt + \int_{\Delta t} \left( W(d\Sigma_1 - d\Sigma_2) \right),
\]

where \( d\Sigma_i := \sigma_i dt + \Sigma_i d\eta, d\Lambda_i := \lambda_i dt + \Lambda_i d\eta \) for \( i \in \{1, 2\} \). It proves useful to introduce the quantities \( \dot{\Sigma}_{\Delta t}(t) = \frac{1}{\Delta t} \int_{\Delta t} (d\Sigma_1 - d\Sigma_2) \) and \( \dot{\Lambda}_{\Delta t}(t) = \frac{1}{\Delta t} \int_{\Delta t} (d\Lambda_1 - d\Lambda_2) \), which can be regarded as the average constraint forces of the error dynamics over the time lapse \( \Delta t \).

Subsequently, we take the limit \( t \to \infty \) and use \( v(t) \to 0 \) and \( e \to c \) for \( t \to \infty \). The integrated equality of measures, divided by \( \Delta t \), yields

\[
\begin{pmatrix}
0 \\
0
\end{pmatrix} = \left( 0 \right) + \lim_{t \to \infty} \begin{pmatrix}
W\Sigma_{\Delta t}(t) \\
W\Lambda_{\Delta t}(t)
\end{pmatrix},
\]

which describes, in an averaged sense, the equilibrium of forces at infinity. Since the columns \( w_i \) of \( W \) are linearly independent, we obtain

\[
Kc = \lim_{t \to \infty} \sum_{i=1}^m w_i \dot{\Lambda}_{\Delta t,i}(t),
\]

which is the desired result.
from which we deduce that each of the limits \( \lim_{t \to \infty} \tilde{\Lambda}_{\Delta t,i}(t) \) has to exist. Let \( I \) be the set of constraints for which the limit \( \lim_{t \to \infty} \tilde{\Lambda}_{\Delta t,i}(t) \) does not vanish for any \( \Delta t \). The error in constraint distances for these constraints vanish due to the force and impact laws (10)–(11), i.e. \( g_{1,I} - g_{2,I} = W_I^T c = 0 \). We multiply equation (21) from the left by \( W_I^T K^{-1} \) and obtain

\[
0 = W_I^T c = W_I^T K^{-1} \lim_{t \to \infty} \sum_{i \in I} w_i \tilde{\Lambda}_{\Delta t,i}(t) = L_I \lim_{t \to \infty} \tilde{\Lambda}_{\Delta t,I}(t),
\]

where the notation introduced in Proposition 1 has been used for \( L_I \) and \( \tilde{\Lambda}_{\Delta t,I} \). The positive definiteness of \( L_I \) yields \( \lim_{t \to \infty} \tilde{\Lambda}_{\Delta t,I}(t) = 0 \) and we obtain \( c = 0 \). Therefore, the equilibrium at the origin \( \begin{pmatrix} e \\ v \end{pmatrix} = 0 \) is globally uniformly attractively stable, which concludes the proof. \( \square \)

Theorem 1 together with Proposition 1 imply that any solution of the slave system approaches the solution of the master system and can thus be used as state estimate.

4 Examples

In Section 3 we have presented a new observer design for the class of mechanical systems shown in Section 2. The proposed observer uses only the information which constraints of the observed system are open or closed and no continuous measurement of the states is necessary. The synchronization of the unilaterally coupled systems is based on the attractive incremental stability of the observer dynamics. The results are illustrated using simulations of two examples of impact oscillators.

**Example 1 (Double mass impact oscillator).** The first example is a double mass impact oscillator, which consists of two masses coupled by spring-damper elements and the upper mass is harmonically forced. Figure 1 depicts the synchronization setup with the coupled master (left) and slave system (right). Both systems are described by (7)–(11) together with (2)–(3) subjected to switched geometric unilateral constraints. The switching functions \( \chi(t) \) are generated by the constraint distances \( g_{m} \) of the master system. Therefore, the (real) master system is subjected to geometric unilateral constraints, whereas the (artificial) slave system is a perfect replica subjected to switched geometric unilateral constraints that are switched on when the corresponding constraints of the master system are closed. The generalized Poisson’s impact law is chosen with a global coefficient of restitution of \( \varepsilon = 0.8 \).

The system matrices \( M, C \) and \( K \) are all symmetric and positive definite. Assumption A1 is fulfilled, since the constraint distances of the master system are absolutely continuous. The external forcing is chosen as a harmonic function such that Assumption A2 is fulfilled. The impact law is described by a monotone impact map, which fulfills Assumption A3. Since the Assumptions A1–A3 from Section 3 are fulfilled, the slave system is attractively incrementally stable according to Theorem 1. Therefore, the synchronization error tends to zero and the slave system reproduces the full state of the master system using only the impact time instants.

The example is simulated for a certain choice of parameters, and the Lyapunov function (16) is shown in Figure 2 for the observer presented in this paper with position jumps (blue) as well as for the observer without position jumps presented in [4] (red). Both observers tend to the observed system, but the observer including position jumps has a higher synchronization rate.

Figure 3 shows the time evolution of the distance between the ground and the first mass. Depicted is the master system (black), the slave system including position jumps (blue)
Figure 1. Master-slave system unidirectionally coupled by the Boolean switching functions $\chi(t)$.

Figure 2. Lyapunov function with position jumps (blue) and without position jumps (red).

Figure 3. Time evolution of the generalized coordinate and velocity of the first mass of the double mass impact oscillator for the master system (black), slave system with position jumps (blue) and without position jumps (red).

and the slave system without position jumps (red). The newly proposed observer tends faster to the master system since it additionally imposes a projection of the generalized coordinates. Both observers reproduce the full state of the observed system using only the impact time information also in the presence of accumulation points, which correspond to infinitely many impacts in a finite time interval (also called Zeno behavior).

Example 2 (Chain of rocking blocks). The second example consists of a chain of blocks which are connected by spring-damper elements as depicted in Figure 4. Each block can move vertically and rotate about its center of gravity. The lowest block is connected to the harmonically moving ground. Between each neighboring block and between the lowest block and the ground there are on both sides unilateral constraints. Only small deflections are considered such that the system can be considered linear time-invariant with symmetric and positive definite system matrices. The generalized Poisson’s impact law is chosen with a global coefficient of restitution of $\varepsilon = 0.9$. The master system is subjected to geometric unilateral constraints and the slave system is subjected to switched geometric unilateral constraints. The slave system is a.i.s. according to Theorem 1.
The coupled system is simulated with a chain consisting of 10 blocks, which corresponds to 20 degrees of freedom. As for Example 1, the observer presented in this paper with position jump as well as the observer without position jumps presented in [4] are implemented. The time evolution of the constraint distance and coordinate of a constraint (bottom right constraint in Figure 4) is shown in Figure 5 for the master system (black), slave system with position jumps (blue) and without position jumps (red). The Lyapunov function for both observers is depicted in Figure 6 using a logarithmic scale. During the time interval $4 \leq t \leq 8$, the Boolean switching functions $\chi(t)$ are generated by the master system, and the slave system (having switched constraints) acts as a state observer. For illustrative purposes, the switching functions are generated by the slave system itself for $t \in [0, 4)$. Therefore, the slave systems are subjected to geometric unilateral constraints during this time interval and there is only one solution visible since the solutions of the uncoupled slave systems are identical.

The slave systems are initialized using initial conditions which are very close to the initial conditions of the master system. The uncoupled case shows that the master and the slave system do generally not synchronize without any coupling. Furthermore, the system has extreme sensitivity on initial conditions such that the solutions will diverge for any small initial error. Therefore, there is no local synchronization and the zero-solution of the error dynamics is unstable in the uncoupled case.

5 Conclusions

An observer is proposed which uses the new concept of switched geometric unilateral constraints in order to extend the approach presented in [4]. These constraints introduce position jumps which are feasible since the observer is not a physical system. The new observer clearly improves the synchronization speed and it relaxes the assumptions on the switching functions, e.g. it allows for persistent constraints (constraints which close and do not open again).

The presented observer uses the property of attractive incremental stability and is based on master–slave synchronization. The observer uses only the information of the impact time instants and no continuous measurement is used. Furthermore, it reproduces the full state of the observed system for every initial condition and also in the presence of accumulation points.
Figure 5. Time evolution of the constraint distance and coordinate of a constraint (bottom right in Figure 4) for the master system (black), slave system with position jumps (blue) and without position jumps (red). The systems are decoupled during the time interval $0 \leq t < 4$ for which there is generally no (local) synchronization.

Figure 6. Lyapunov function for the coupled chain of rocking blocks system with position jumps (blue) and without position jumps (red) using a logarithmic scale.

ACKNOWLEDGMENT
This research is supported by the Swiss National Science Foundation through the project ‘Synchronization of Dynamical Systems with Impulsive Motion’ (SNF 200021-144307).

REFERENCES


Towards a Maximal Monotone Impact Law for Newton’s Cradle

Tom Winandy, Remco I. Leine

Institute for Nonlinear Mechanics
University of Stuttgart
Pfaffenwaldring 9, D-70569 Stuttgart, Germany
[winandy, leine]@inm.uni-stuttgart.de

ABSTRACT
The 3-ball Newton’s cradle is used as a stepping stone to divulge the structure of impact laws. A continuous cone-wise linear impact law which maps the pre-impact contact velocities to the post-impact contact velocities is proposed for the 3-ball Newton’s cradle. The proposed impact law is kinematically, kinetically, and energetically consistent. It reproduces the outcomes of experimental observation. Moreover, it is in accordance with the outcome of the collision of three identical linear-elastic thin rods for which the impact process is governed by the one-dimensional wave equation. The proposed impact law is shown to be non-expansive. Therefore, the relationship between the mean contact velocity and its dual, the impulsive force, is maximal monotone. A counter-example to maximal cyclical monotonicity of this relationship allows to conclude that no dissipation function exists for the proposed impact law.

Keywords: Newton’s cradle, unilateral constraints, impact, convex analysis, wave equation.

1 INTRODUCTION
In this paper, we present an impact law for Newton’s cradle with 3 balls. We construct a continuous cone-wise linear impact law which is non-dissipative. The impact law provides the classical outcomes of the 3-ball Newton’s cradle. The 3-ball Newton’s cradle can be modelled as three identical linear-elastic thin rods which collide. The outcomes of the impact law are in accordance with the results of this model, which is governed by the one-dimensional wave equation.

Our aim is to divulge the structure of impact laws in order to be able to formulate maximal monotone impact laws for rigid multi-body systems that do not have the problems of existing impact laws such as kinematic, kinetic, and energetic inconsistency [1]. It is interesting to consider Newton’s cradle because its phenomena cannot be described by the classical Newton’s or Poisson’s instantaneous impact law (see Figure 1(b)).

The impenetrability of unilateral constraints, which requires that the post-impact contact velocities \( \gamma^+ \) are non-negative, is referred to in this paper as kinematic consistency. In view of numerical integration, an impact-law should guarantee that arbitrary (also kinematically inadmissible) pre-impact contact velocities are mapped to kinematically admissible post-impact contact velocities.

Finally, we show that the impact law enjoys the maximal monotonicity property. The interest in the maximal monotonicity property stems from stability analysis and control of mechanical systems with unilateral constraints [2]. The maximal monotonicity property often allows the formulation of Lyapunov-based stability statements. In [3], the maximal monotonicity property is used for the design of state observers for unilaterally constrained multibody systems. Since the maximal monotonicity property implies dissipativity it might be a physically reasonable property for an impact law.

2 THE 3-BALL NEWTON’S CRADLE
The 3-ball Newton’s cradle is shown in Figure 1(a). It consists of three balls of equal mass \( m \) with horizontal positions \( \mathbf{q} = (q_1, q_2, q_3)^T \) and velocities \( \dot{\mathbf{q}} = (u_1, u_2, u_3)^T \). The contact distances are given by \( \mathbf{g} = (q_2 - q_1, q_3 - q_2)^T \). The unilateral constraint \( \mathbf{g} \geq 0 \) expresses the fact that contacts
can open but that the balls may not penetrate each other. The contact velocities are given by the relative velocities between the balls \( \gamma = (\gamma_1, \gamma_2)^T = (u_2 - u_1, u_3 - u_2)^T \). The pre- and post-impact velocities are designated by \( u^- \) and \( u^+ \), respectively. Analogously, \( \gamma^- \) and \( \gamma^+ \) designate the pre- and post-impact contact velocities.

The impact equations of the system can be written in the following matrix form

\[
M(u^+ - u^-) = WA,
\]

\[
\gamma^\pm = W^T u^\pm,
\]

where \( A = (A_1, A_2)^T \) are the impulsive contact forces during the impact. The impulsive force \( A_1 \) acts between balls 1 and 2, while \( A_2 \) occurs between balls 2 and 3. The matrix \( W \) is the matrix of generalized force directions for which holds \( W^T = \partial g / \partial q \). For the 3-ball Newton’s cradle, the mass matrix \( M \) and the matrix of generalized force directions \( W \) are

\[
M = \begin{pmatrix}
  m & 0 & 0 \\
  0 & m & 0 \\
  0 & 0 & m
\end{pmatrix}
\]

and

\[
W = \begin{pmatrix}
  -1 & 0 \\
  1 & -1 \\
  0 & 1
\end{pmatrix}.
\]

The impact equation (1) needs to be complemented by an impact law that has the mathematical structure of a set-valued relationship \( [4] \)

\[
-A \in \mathcal{H}(\tilde{\gamma}),
\]

where

\[
\tilde{\gamma} = \frac{1}{2}(\gamma^+ + \gamma^-).
\]

The operator \( \mathcal{H} : \mathbb{R}^n \rightarrow \mathbb{R}^n \) is in general a set-valued operator. The combination of (1) and (2) yields the impact equation in contact velocities

\[
\gamma^+ - \gamma^- = GA \quad \text{with} \quad G := W^T M^{-1} W.
\]

The matrix \( G \) is referred to as the Delassus operator.

Alternatively to the formulation as a set-valued relationship (4), the impact law can be expressed by a mapping \( S \) from pre- to post-impact contact velocities

\[
\gamma^+ = S(\gamma^-),
\]

**Figure 1.** Left: Newton’s cradle with 3 balls of mass \( m \). Right: An example of outcomes by Newton’s and Poisson’s impact law.
or by a mapping \( Z \) from pre- to post-impact generalized velocities

\[
\mathbf{u}^+ = Z(\mathbf{u}^-). 
\]  

(8)

An impact law should be kinematically, kinetically, and energetically consistent:

- Pre-impact contact velocities \( \mathbf{\gamma}^- \) and post-impact contact velocities \( \mathbf{\gamma}^+ \) are called **kinematically admissible** or **kinematically consistent** if

\[
\mathbf{\gamma}^- \leq \mathbf{0} \quad \text{and} \quad \mathbf{\gamma}^+ \geq \mathbf{0},
\]  

(9)

respectively.

- **Kinetic consistency** is required by the unilateral character of non-adhesive contacts which requires the contact forces to be non-negative

\[
\Lambda \geq \mathbf{0}. 
\]  

(10)

The contact force vanishes if the contact is open, i.e. if \( \mathbf{g} > \mathbf{0} \). If \( \mathbf{g} = \mathbf{0} \) the contact is closed and it can only transfer non-negative contact forces.

- **Energetic consistency** means that there is no increase in energy during the impact. Let the kinetic energy before and after the impact be designated by

\[
T^- = \frac{1}{2} \mathbf{u}^-\mathbf{T} \mathbf{u}^- \quad \text{and} \quad T^+ = \frac{1}{2} \mathbf{u}^+\mathbf{T} \mathbf{u}^+,
\]  

respectively. Energetic consistency then requires that

\[
T^+ \leq T^- \iff T^+ - T^- \leq 0,
\]  

(11)

which can be expressed in terms of pre- and post impact velocities

\[
\mathbf{u}^+\mathbf{T} \mathbf{u}^+ - \mathbf{u}^-\mathbf{T} \mathbf{u}^- = (\mathbf{u}^+ + \mathbf{u}^-)^\mathbf{T} \mathbf{M} (\mathbf{u}^+ + \mathbf{u}^-) \leq 0.
\]  

(12)

The use of (1), (2), and (6) permits to rewrite (12) as

\[
(\mathbf{\gamma}^+ + \mathbf{\gamma}^-)^\mathbf{T} \mathbf{G}^{-1} (\mathbf{\gamma}^+ - \mathbf{\gamma}^-) = \mathbf{\gamma}^+\mathbf{T} \mathbf{G}^{-1} \mathbf{\gamma}^+ - \mathbf{\gamma}^-\mathbf{T} \mathbf{G}^{-1} \mathbf{\gamma}^- \leq 0.
\]  

(13)

The conditions (12) and (13) for energetic consistency can be expressed using the norms with metric \( \mathbf{M} \) and \( \mathbf{G}^{-1} \), respectively

\[
||\mathbf{u}^+||^2_\mathbf{M} \leq ||\mathbf{u}^-||^2_\mathbf{M} \quad \text{and} \quad ||\mathbf{\gamma}^+||^2_{\mathbf{G}^{-1}} \leq ||\mathbf{\gamma}^-||^2_{\mathbf{G}^{-1}}.
\]  

(14)

From [5, 4], it is known that the maximal monotonicity of the operator \( \mathcal{H} \) in (4) is equivalent to non-expansivity properties of the impact mappings (7) and (8).

**Definition 1** (Maximal monotonicity [5]). A mapping \( \mathcal{T} : \mathbb{R}^n \rightrightarrows \mathbb{R}^n \) is called monotone if it has the property that

\[
(\mathbf{y}_A - \mathbf{y}_B)^\mathbf{T}(\mathbf{x}_A - \mathbf{x}_B) \geq 0,
\]  

(15)

whenever \( \mathbf{y}_A \in \mathcal{T}(\mathbf{x}_A), \mathbf{y}_B \in \mathcal{T}(\mathbf{x}_B) \). Moreover, \( \mathcal{T} \) is called maximal monotone if it is monotone and its graph cannot be enlarged without destroying this property.

**Definition 2** (Non-expansivity [5, 4]). A mapping \( \mathcal{F} : \mathbb{R}^n \rightrightarrows \mathbb{R}^n \) is called non-expansive in the metric \( \mathbf{P} \) if it has the property that

\[
||\mathbf{y}_A - \mathbf{y}_B||_\mathbf{P} \leq ||\mathbf{x}_A - \mathbf{x}_B||_\mathbf{P},
\]  

(16)

whenever \( \mathbf{y}_A \in \mathcal{F}(\mathbf{x}_A), \mathbf{y}_B \in \mathcal{F}(\mathbf{x}_B) \).
Furthermore, the set-valued operator $\mathcal{H}$ can be written as the subdifferential to a convex proper lower semicontinuous (l.s.c.) dissipation function $\Phi$ such that

$$-\Lambda \in \mathcal{H}(\tilde{\gamma}) = \partial \Phi(\tilde{\gamma}),$$

if and only if $\mathcal{H}$ is maximal cyclically monotone.

**Definition 3** (Cyclical monotonicity [5]). A mapping $T: \mathbb{R}^n \mapsto \mathbb{R}^n$ is cyclically monotone if for any cycle of $m$ points $x_A, x_B, \ldots, x_Z$ (for arbitrary $m \geq 2$) and elements $y_i \in T(x_i)$, one has

$$y_A^T(x_B - x_A) + y_B^T(x_C - x_B) + \ldots + y_Z^T(x_A - x_Z) \leq 0.$$  

(18)

It is maximal cyclically monotone if it is cyclically monotone and its graph cannot be enlarged without destroying this property.

Note that cyclical monotonicity is a stronger condition than monotonicity. Definition 3 reduces to Definition 1 when $m = 2$. The relations between the non-expansivity and monotonicity properties are shown in Figure 2.

3 THE SEQUENTIAL IMPACT LAW

We propose a continuous cone-wise linear impact mapping $S: \mathbb{R}^2 \to \mathbb{R}^2$, $\gamma^- \mapsto \gamma^+$ for the 3-ball Newton’s cradle. The impact mapping $S$ takes the form

$$\gamma^+ = S(\gamma^-) = Q_i \gamma^-,$$

(19)

where $Q_i \in \mathbb{R}^{2 \times 2}$ are 2-by-2 matrices which apply in a corresponding cone in the $(\gamma_1^-, \gamma_2^-)$-plane. We construct the matrices $Q_i$ together with their respective cones $C_i$ by demanding the following properties of the impact law:

**P1** The mapping is continuous, i.e. $Q_i v_i = Q_{i+1} v_i$ with $v_i$ being the direction of the boundary half-line between the cones $C_i$ and $C_{i+1}$.

**P2** Conservation of energy holds, i.e. $\|\gamma^-\|_{G^{-1}} = \|Q_i \gamma^-\|_{G^{-1}} = \|\gamma^-\|_{G^{-1}}$ for all matrices $Q_i$. This implies energetic consistency.

**P3** Each cone $Q_i$ is mapped to the entire first quadrant, i.e. the cone $C_i$ is spanned by the columns of $Q_i^{-1}$. This implies kinematic consistency.
We start with the first quadrant (see Figure 3(a)). Pre-impact contact velocities from the first quadrant are positive which means that no impact occurs. Therefore, we set

$$Q_I = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 \end{pmatrix},$$

which means that $\gamma^+ = \gamma^-$ for all $\gamma^- \in C_I$. The cone $C_I$ is spanned by the columns of $Q_I^{-1}$.

Next, we proceed to the cone $C_{IIa}$ on the left of $C_I$ as shown in Figure 3(b). The boundary between the two cones is given by the positive $\gamma^-$-axis. Continuity and conservation of energy ($P1$ and $P2$) lead us to the matrix

$$Q_{IIa} = \begin{pmatrix} -1 & 0 \\ 1 & 1 \end{pmatrix}.$$  

The direction of the boundary to the next cone can be read from $Q_{IIa}^{-1}$ ($P3$) as it is shown in Figure 3(b), such that

$$v_{IIa} = \begin{pmatrix} -1 \\ 1 \end{pmatrix}.$$  

We can proceed analogously to find all six cones $C_i$ with $i \in \{I, IIa, IIb, III, IVa, IVb\}$ together with their corresponding matrices

$$Q_I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad Q_{IIa} = \begin{pmatrix} -1 & 0 \\ 1 & 1 \end{pmatrix}, \quad Q_{IIb} = \begin{pmatrix} 0 & 1 \\ -1 & -1 \end{pmatrix}, \quad Q_{III} = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}, \quad Q_{IVa} = \begin{pmatrix} 1 & 1 \\ 0 & -1 \end{pmatrix}, \quad Q_{IVb} = \begin{pmatrix} -1 & -1 \\ 1 & 0 \end{pmatrix}.$$  

The repartition of the $(\gamma^-, \gamma^-)$-plane into the six cones $C_i$ with $i \in \{I, IIa, IIb, III, IVa, IVb\}$ is depicted in Figure 4(a). The symmetry of the problem appears in the symmetry between the matrices $Q_{IIa}$ and $Q_{IVa}$ as well as between $Q_{IIb}$ and $Q_{IVb}$.

In the following, we will call the impact law (19) the Sequential Impact Law because it is equivalent to a sequence of impacts between only two balls. This can be seen by the following properties

$$Q_{IIb} = Q_{IVa} Q_{IIa}, \quad Q_{IVb} = Q_{IIa} Q_{IVa}, \quad Q_{III} = Q_{IVa} Q_{IIa} Q_{IVa} = Q_{IIa} Q_{IVa} Q_{IIa},$$

where $Q_{IIa}$ and $Q_{IVa}$ describe the impact between only two of the three balls as will be shown below.
Figure 4. Left: The different cones in the \((\gamma_1^-, \gamma_2^-)\)-plane. The dot-dashed line marks the symmetry line. Right: Idealized observations.

After having derived the Sequential Impact Law, we want to argue why it is a reasonable choice. In the following, the implications of the Sequential Impact Law are discussed for the four quadrants of the \((\gamma_1^-, \gamma_2^-)\)-plane.

The first quadrant is equal to the cone \(C_I\) and it corresponds to two positive pre-impact contact velocities such that no impact happens. The identity map is the only reasonable choice for the first quadrant.

The third quadrant is equal to the cone \(C_{III}\) and it corresponds to both pre-impact contact velocities being negative and therefore kinematically admissible. Hence, the third quadrant contains all the classical experimental outcomes which can be realized with a 3-ball Newton’s cradle. Figure 4(b) gives three examples of idealized observations from cone \(C_{III}\). The Sequential Impact Law provides these idealized experimental outcomes.

The second and the fourth quadrant correspond to one pre-impact contact velocity being positive and the other being negative. Each one of these quadrants contains two different cones, because the magnitude of the positive pre-impact contact velocity determines whether the corresponding outer ball participates in the impact process or not. The cones \(C_{IIa}\) and \(C_{IVa}\), which are adjacent to the first quadrant, correspond to a single impact between only two of the three balls. In the cone \(C_{IIa}\), the right ball does not participate in the impact process because it has a positive pre-impact contact velocity that prevents it from colliding with the middle ball. This can be seen by considering the impact equation in the contact velocities (6) and the Sequential Impact Law (19) for pre-impact velocities

\[
\gamma^+ - \gamma^- = (Q_{Hb} - I) \gamma^- = G \mathbf{A},
\]

where \(I\) denotes the identity matrix. Eq. (25) yields the impulsive force

\[
\mathbf{A} = \begin{pmatrix} -m\gamma_1^- \\ 0 \end{pmatrix},
\]

from which it becomes apparent that the right ball does not participate in the impact process. In \(C_{IVa}\), it is the left ball that is not subjected to any impact. For pre-impact velocities belonging to the cones \(C_{IIb}\) and \(C_{IVb}\), the Sequential Impact Law provides the same result as it is given by the generalized Newton and by the generalized Poisson impact law for the non-dissipative impact of two balls [6, 7]. The positive pre-impact contact velocity in the cones \(C_{IIb}\) and \(C_{IVb}\) does not prevent the interaction between the three balls through wave effects.

The third quadrant basically completely describes the physics of Newton’s cradle as it covers all physically realizable experiments with Newton’s cradle. Nevertheless, the first, the second, and the
fourth quadrant are needed to deal with kinematically inadmissible pre-impact contact velocities, which is important in view of numerical simulation. Small numerical errors may lead to positive pre-impact contact velocities and an impact law should map these pre-impact contact velocities to physically reasonable post-impact contact velocities.

In the next section, we provide a further argument for the validity of the Sequential Impact Law by showing that it provides the same outcomes as the one-dimensional wave equation does for the collision of three identical thin rods.

4 THIN ROD MODEL OF THE 3-BALL NEWTON’S CRADLE

Wave effects play a crucial role in the impact process of the 3-ball Newton’s cradle. We model the system with three identical thin rods (see Figure 5). This leads us to a description of the impact process that is governed by the one-dimensional wave equation.

The rods have cross-section \( A \) and density \( \rho \). We consider the stresses acting on a differential element of the rod as it is shown in Figure 5. The position of the differential element is denoted by \( x \). The displacement field is referred to as \( u(x,t) \). The balance of linear momentum in \( x \) direction for the differential element is then given by

\[
\rho A \, dx \, u_{tt}(x,t) = A \left( \sigma(x+dx,t) - \sigma(x,t) \right). 
\]

(27)

The mass element \( dm \) can be expressed in terms of \( dx \) as

\[
dm = \rho Adx. 
\]

(28)

Further, we assume that the rods behave linear-elastically and thus obey Hooke’s law

\[
\sigma = E \varepsilon = Eu_x, 
\]

(29)

where \( E \) and \( \varepsilon \) designate the Young’s modulus and the strain, respectively. Using (28) and (29), we can rewrite (27) as

\[
\rho Adx \, u_{tt}(x,t) = AD \left( u_x(x+dx,t) - u_x(x,t) \right). 
\]

(30)

Dividing (30) by \( \rho Adx \) and letting \( dx \to 0 \) yields the classical one-dimensional wave equation

\[
u_{tt}(x,t) = c^2 u_{xx}(x,t) \quad \text{with} \quad c^2 = \frac{E}{\rho},
\]

(31)

where \( c \) denotes the propagation velocity of longitudinal waves in the rod.
Our aim is to investigate the impact effects in Newton’s cradle by considering colliding rods. An impact between two colliding rods leads to discontinuities in the velocity and in the strain which expand through the colliding rods with velocity \(c\). These discontinuities can only propagate along characteristics of the solution of the wave equation (31). The construction of a characteristics diagram provides a way to investigate the wave propagation process. Detailed information about waves in elastic solids can be found in \([8, 9]\). In order to be able to construct characteristics diagrams, we first summarize some transition conditions of the longitudinal waves in Table 1.

### 4.1 Collision of two identical thin rods

We consider the collision of two identical thin rods. Before the impact, both rods are undeformed. Initially, the left rod has a uniform velocity \(v\) while the right rod is at rest. We assume that the colliding ends of the rods have the same velocity as soon as they touch. The pre-impact configuration can be seen in Figure 6(a). The initial conditions can be stated as

\[
\begin{align*}
  u_x(x, 0) &= 0 & \text{if } x \in [0, 2l], \\
  u_t(x, 0) &= \begin{cases} 
  v & \text{if } x \in [0, l], \\
  0 & \text{if } x \in (l, 2l].
\end{cases}
\end{align*}
\]

Discontinuities in the velocity and in the strain can only propagate along their characteristics. These characteristics of the wave equation (31) delimit regions inside which the strains and the velocities are constant. Therefore, we state the strain and velocity for each region in the characteristics diagram in Figure 6(a). The transition properties from Table 1 allow the construction of the characteristics diagram and the calculation of the corresponding strains and velocities which are prevalent in the different regions in the characteristics diagram

\[
\begin{align*}
  \text{Region (1):} & \quad u_{1,x} = 0, \quad u_{1,t} = v, \\
  \text{Region (2):} & \quad u_{2,x} = 0, \quad u_{2,t} = 0, \\
  \text{Region (3):} & \quad u_{3,x} = -\frac{v}{2c}, \quad u_{3,t} = \frac{v}{2}, \\
  \text{Region (4):} & \quad u_{4,x} = 0, \quad u_{4,t} = 0, \\
  \text{Region (5):} & \quad u_{5,x} = 0, \quad u_{5,t} = v.
\end{align*}
\]

In A, the contact opens, the left rod is at rest and the right rod has the uniform velocity \(v\).
We can conclude that the collision of two identical thin rods leads to the same result as Newton’s and Poisson’s impact law do for two balls in the non-dissipative case. Moreover, the Sequential Impact Law also provides this outcome in the cones $C_{Ia}$ and $C_{IVa}$ which correspond to the present situation of a single impact between two balls.

4.2 Collision of three identical thin rods

We now consider three identical thin rods. Initially, the middle rod is at rest. It is approached by the left and by the right rod which have velocities $2v$ and $-v$, respectively. Again, the rods are undeformed before the collision. The initial conditions can be written as

\[
 u_x(x,0) = 0 \quad \text{if } x \in [0,3l],
\]

\[
 u_t(x,0) = \begin{cases} 
 2v & \text{if } x \in [0,l], \\
 0 & \text{if } x \in (l,2l], \\
 -v & \text{if } x \in (2l,3l].
\end{cases}
\]

As in the previous case, we assume that the colliding ends of the rods have the same velocity as soon as they touch. Again, a characteristics diagram is constructed (see Figure 6(b)) using the transition properties from Table 1. The following velocities and strains are obtained for the different regions in the diagram.

<table>
<thead>
<tr>
<th>Region</th>
<th>$u_{1x}$</th>
<th>$u_{1t}$</th>
<th>$u_{8x}$</th>
<th>$u_{8t}$</th>
<th>$u_{9x}$</th>
<th>$u_{9t}$</th>
<th>$u_{10x}$</th>
<th>$u_{10t}$</th>
<th>$u_{11x}$</th>
<th>$u_{11t}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>$2v$</td>
<td>0</td>
<td>0</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>0</td>
<td>$v$</td>
</tr>
<tr>
<td>2</td>
<td>$v$</td>
<td>0</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>$v$</td>
<td>0</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>0</td>
<td>$v$</td>
</tr>
<tr>
<td>3</td>
<td>$v$</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>$2v$</td>
<td>0</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>$2v$</td>
<td>0</td>
<td>$v$</td>
</tr>
<tr>
<td>4</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>$v$</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>$2v$</td>
<td>0</td>
<td>$v$</td>
</tr>
<tr>
<td>5</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>$v$</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>$2v$</td>
<td>0</td>
<td>$v$</td>
</tr>
<tr>
<td>6</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>$v$</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>$2v$</td>
<td>0</td>
<td>$v$</td>
</tr>
<tr>
<td>7</td>
<td>$v$</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>$2v$</td>
<td>0</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>$2v$</td>
<td>0</td>
<td>$v$</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>$2v$</td>
<td>0</td>
<td>0</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>0</td>
<td>$v$</td>
</tr>
<tr>
<td>9</td>
<td>$v$</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>$2v$</td>
<td>0</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>$2v$</td>
<td>0</td>
<td>$v$</td>
</tr>
<tr>
<td>10</td>
<td>$v$</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>$2v$</td>
<td>0</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>$2v$</td>
<td>0</td>
<td>$v$</td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>$2v$</td>
<td>0</td>
<td>0</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>0</td>
<td>$v$</td>
</tr>
<tr>
<td>12</td>
<td>$v$</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>$2v$</td>
<td>0</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>$2v$</td>
<td>0</td>
<td>$v$</td>
</tr>
<tr>
<td>13</td>
<td>0</td>
<td>$2v$</td>
<td>0</td>
<td>0</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>$-v$</td>
<td>$\frac{v}{2}$</td>
<td>0</td>
<td>$v$</td>
</tr>
</tbody>
</table>

Figure 6. Characteristics diagrams for colliding identical thin rods.
In $B$ and $C$, both contacts open simultaneously. The left rod has a post-impact velocity of $v$. The middle rod is at rest and the right rod has a post-impact velocity of $2v$. This corresponds exactly to the outcome which is provided by the Sequential Impact Law as can be seen by calculating the pre- and post-impact relative velocities of the rods

$$\gamma^- = \begin{pmatrix} -2v \\ -v \end{pmatrix} \quad \text{and} \quad \gamma^+ = \begin{pmatrix} v \\ 2v \end{pmatrix},$$

for which indeed holds $\gamma^+ = S(\gamma^-) = \Sigma_{11} \gamma^-$. Note that Newton’s and Poisson’s instantaneous impact law would give a different outcome.

## 5 CONTRACTION PROPERTIES OF THE SEQUENTIAL IMPACT LAW

Since we are interested in the mathematical structure of impact laws, we investigate which properties from Figure 2 hold for the Sequential Impact Law.

### 5.1 Non-expansivity of the Sequential Impact Law

In the following, we show that the Sequential Impact Law is non-expansive in the metric $G^{-1}$. This means that the set-valued relationship (4) between the dual variables $\gamma$ and $A$ is maximal monotone.

**Theorem 1.** The impact mapping (19) is non-expansive in the metric $G^{-1}$, i.e.

$$\|\gamma^+ - \gamma^-\|_{G^{-1}} \leq \|\gamma - \gamma^+\|_{G^{-1}} \quad \forall \gamma^+, \gamma \in \mathbb{R}^2.$$

**Proof.** The condition (37) needs to hold for arbitrary pairs of pre-impact contact velocities $\gamma^+_A$ and $\gamma^+_B$. The idea behind the proof is to decompose the line that connects the points $\gamma^+_A$ and $\gamma^+_B$ into a series of segments which lie in a single cone respectively. This decomposition can be done using telescopic expansion

$$\|\gamma^+_A - \gamma^+_B\|_{G^{-1}} = \|\gamma^+_A - \gamma^+_1 + \gamma^+_1 - \gamma^+_2 + \ldots + \gamma^+_k - \gamma^+_B\|_{G^{-1}},$$

where the $\gamma^+_i$ are the images of $\gamma^+_i$ which lie on the boundaries between the cones. Figure 7(a) shows an example of this decomposition. At the boundary between the cones $C_i$ and $C_{i+1}$, we have that $\gamma^+_i = Q_i \gamma^+_i = Q_{i+1} \gamma^+_i$ due to continuity (P1). Therefore, eq. (38) can be written as

$$\|\gamma^+_A - \gamma^+_B\|_{G^{-1}} = \|Q_A(\gamma^+_A - \gamma^+_i) + Q_1(\gamma^+_i - \gamma^+_2) + \ldots + Q_B(\gamma^+_k - \gamma^+_B)\|_{G^{-1}}.$$

From the triangle inequality, it follows that

$$\|\gamma^+_A - \gamma^+_B\|_{G^{-1}} = \|Q_A(\gamma^+_A - \gamma^+_i) + Q_1(\gamma^+_i - \gamma^+_2) + \ldots + Q_B(\gamma^+_k - \gamma^+_B)\|_{G^{-1}}$$

$$\leq \|Q_A(\gamma^+_A - \gamma^+_i)\|_{G^{-1}} + \|Q_1(\gamma^+_i - \gamma^+_2)\|_{G^{-1}} + \ldots + \|Q_B(\gamma^+_k - \gamma^+_B)\|_{G^{-1}}.$$  

Due to the energy conservation property P2, it holds that $\|Q_A \gamma^+_i\|_{G^{-1}} = \|\gamma^+_i\|_{G^{-1}}$. This leads to

$$\|\gamma^+_A - \gamma^+_B\|_{G^{-1}} \leq \|\gamma^+_A - \gamma^+_1\|_{G^{-1}} + \|\gamma^+_1 - \gamma^+_2\|_{G^{-1}} + \ldots + \|\gamma^+_k - \gamma^+_B\|_{G^{-1}}.$$  

(41)

Each $\gamma^+_i$ can be expressed in terms of $\gamma^+_A$ and $\gamma^+_B$ as the convex combination

$$\gamma^+_i = \alpha_i \gamma^+_A + (1 - \alpha_i) \gamma^+_B \quad \text{with} \quad \alpha_i \in [0,1].$$

(42)

Eq. (42) allows to rewrite the terms of the right-hand side of (41) in the following way

$$\|\gamma^+_A - \gamma^+_1\|_{G^{-1}} = (1 - \alpha_1)\|\gamma^+_A - \gamma^+_B\|_{G^{-1}},$$

$$\|\gamma^+_1 - \gamma^+_2\|_{G^{-1}} = (\alpha_1 - \alpha_2)\|\gamma^+_A - \gamma^+_B\|_{G^{-1}},$$

$$\vdots$$

$$\|\gamma^+_k - \gamma^+_B\|_{G^{-1}} = \alpha_k\|\gamma^+_k - \gamma^+_B\|_{G^{-1}}.$$  

(43)
Finally, we use the expressions from (43) to rewrite (41) as
\[
\|\gamma_A^+ - \gamma_B^-\|_{G^{-1}} \leq (1 - \alpha_1)\|\gamma_A - \gamma_B^-\|_{G^{-1}} + (\alpha_1 - \alpha_2)\|\gamma_A - \gamma_B^-\|_{G^{-1}} + \ldots + \alpha_k\|\gamma_k - \gamma_B^-\|_{G^{-1}}
\]
\[
= \|\gamma_A - \gamma_B^-\|_{G^{-1}},
\]
which completes the proof.

5.2 A counter-example to maximal cyclical monotonicity

In order to give a counter-example to the maximal cyclical monotonicity of (4), we propose to consider an ABC-cycle which provides a counter-example to the maximal cyclical monotonicity of the impact mapping $S$.

Finally, we use the expressions from (43) to rewrite (41) as
\[
\|\gamma_A^+ - \gamma_B^-\|_{G^{-1}} \leq (1 - \alpha_1)\|\gamma_A - \gamma_B^-\|_{G^{-1}} + (\alpha_1 - \alpha_2)\|\gamma_A - \gamma_B^-\|_{G^{-1}} + \ldots + \alpha_k\|\gamma_k - \gamma_B^-\|_{G^{-1}}
\]
\[
= \|\gamma_A - \gamma_B^-\|_{G^{-1}},
\]
which completes the proof.

The cycle of pre-impact contact velocities (45) is shown in Figure 7(b). The Sequential Impact Law (19) leads to the following post-impact contact velocities
\[
\gamma_A^+ = (v \quad v)^T, \quad \gamma_B^- = (0 \quad v)^T, \quad \gamma_C^- = (0 \quad 0)^T.
\]
\[
\gamma_A^+ = (v \quad v)^T, \quad \gamma_B^- = (0 \quad v)^T, \quad \gamma_C^- = (0 \quad 0)^T.
\]
\[
\gamma_A^+ = (v \quad v)^T, \quad \gamma_B^- = (0 \quad v)^T, \quad \gamma_C^- = (0 \quad 0)^T.
\]

Using (5), (45), and (46), we obtain
\[
\Lambda_A = (2mv \quad mv)^T, \quad \Lambda_B = (mv \quad mv)^T, \quad \Lambda_C = (0 \quad 0)^T.
\]
\[
\Lambda_A = (2mv \quad mv)^T, \quad \Lambda_B = (mv \quad mv)^T, \quad \Lambda_C = (0 \quad 0)^T.
\]
\[
\Lambda_A = (2mv \quad mv)^T, \quad \Lambda_B = (mv \quad mv)^T, \quad \Lambda_C = (0 \quad 0)^T.
\]

Eq. (47) and (48) allow the evaluation of inequality (18) from Definition 3 for the ABC-cycle
\[
\Lambda_A^T(\gamma_B - \gamma_A) + \Lambda_B^T(\gamma_C - \gamma_B) + \Lambda_C^T(\gamma_A - \gamma_C) = -\frac{mv^2}{2} \leq 0.
\]
\[
\Lambda_A^T(\gamma_B - \gamma_A) + \Lambda_B^T(\gamma_C - \gamma_B) + \Lambda_C^T(\gamma_A - \gamma_C) = -\frac{mv^2}{2} \leq 0.
\]
\[
\Lambda_A^T(\gamma_B - \gamma_A) + \Lambda_B^T(\gamma_C - \gamma_B) + \Lambda_C^T(\gamma_A - \gamma_C) = -\frac{mv^2}{2} \leq 0.
\]

Bearing in mind the minus sign from (4), we recognize that (49) is a contradiction to condition (18) from Definition 3. Thus, we can conclude that the set-valued operator $\mathcal{H}$ in (4) is not maximal cyclically monotone for the Sequential Impact Law. Therefore, the Sequential Impact Law cannot be expressed by a convex proper lower semicontinuous dissipation function $\Phi(\gamma)$. 
6 CONCLUSIONS

With the Sequential Impact Law a continuous cone-wise linear impact law has been formulated. It can describe the wave-like phenomena in Newton’s cradle and reproduces the experimental observations. Moreover, the Sequential Impact Law is in accordance with the post-impact velocities provided by the one-dimensional wave equation for the collision of three identical thin rods. The Sequential Impact Law is kinematically, kinetically, and energetically consistent. The impact mapping $S$ of the Sequential Impact Law is non-expansive. Accordingly, the corresponding operator $H$ is maximal monotone. The provided counter-example to maximal cyclical monotonicity lets us conclude that no dissipation function exists for the Sequential Impact Law.

7 ACKNOWLEDGEMENT

This research is supported by the Fonds National de la Recherche, Luxembourg (Project Reference 8864427).

REFERENCES


A numerical study of an autoparametric system with electromagnetic energy harvester

A. Mitura*, K. Kecik*, J. Warminski*, W. Jarzyna#, S. Lenci†

*Department of Applied Mechanics
Lublin University of Technology
Nadbystrzycka 36, 20-618 Lublin, Poland
a.mitura@pollub.pl, k.kecik@pollub.pl, j.warminski@pollub.pl

#Electrical Drives and Electrical Machines Department
Lublin University of Technology
Nadbystrzycka 38, 20-618 Lublin, Poland
w.jarzyna@pollub.pl

†Department of Civil and Building Engineering and Architecture
Polytechnic University of Marche
via Brecce Bianche, 60131 Ancona, Italy
lenci@univpm.it

ABSTRACT
In the paper a strongly nonlinear model of an autoparametric system with added electromagnetic energy harvester is shown. The studied system is considered simultaneously as a device dedicated for vibration absorption and energy harvesting as well. The analysis is limited to the main parametric resonance region. Obtained numerical results show possibility of energy harvesting from different kind of pendulum motion. It is demonstrated that the chaotic behaviour is the best kind of motion for energy harvesting.

Keywords: Autoparametric system, Vibration absorption, Energy harvesting.

1 INTRODUCTION
A pendulum as a nonlinear vibration absorber is used in practice for reducing the high level of vibrations of different engineering structures. For example, the pendulum can be applied in civil structures (high buildings and bridges), which are exposed to wind-induced and seismic vibrations [5]. The main object (which vibration is to be reduced) is often modelled as a simple oscillator. The object with the added pendulum (vibration absorber) is called an autoparametric system. In such systems, the pendulum can perform different types of response [3], for example: an equilibrium state in the upper or lower position (I), rotation (II), swinging (III) or chaotic motion (IV). In the paper [4], a harvester composed of pendulum jointed to the structure was presented. This application gave good results when the pendulum rotated or performed chaotic motion. However, for the vibration reduction, the rotating and chaotic behaviours are unwanted. Therefore, we propose a novel electromagnetic harvesto-absorber system (EMHAS). The EMHAS consists of three magnets (two fixed and one movable) and a coil with an energy receiver located in its electrical circuit. These elements can be mounted inside the pendulum structure. The concept of EMHAS is dedicated for energy harvesting from the pendulum swings (assuming pendulum working as an absorber). The proposed design of the absorber with added harvester is very important, because gives possibility to obtain two effects in the same time: vibration reduction and energy harvesting.

2 DESCRIPTION OF SYSTEM
The studied system, presented in Fig. 1, consists of three basic subsystems. The first is the main object assumed as an oscillator with one degree of freedom composed of mass (M), a linear spring \((k_1)\) and a linear damper \((c)\). The vertical motion of the main object is described by the coordinate \(x\). The oscillator is excited kinematically by the linear spring \((k_2)\), where one of its ends move according to a periodic function \(x_0\). To reduce of vibrations the second subsystem (pendulum) is applied. The motion of the pendulum is described by the angular coordinate \(\varphi\).
The added third subsystem is mounted in the pendulum structure. This electromagnetic energy harvester consists of a movable magnet inside a coil. This moving magnet is “suspended” due to magnetic levitation. It is located between two fixed magnets with polarity configuration: SN-NS-SN. The motion of the magnet is described by the coordinate \( r \) (measured along the pendulum axis). This subsystem allows energy recovery, when the pendulum moves, the movable magnet oscillates and then a voltage is induced.

Creating a mathematical model we take into account mechanical and electrical parts. Moving magnets induces electromotive force \( E_{EM} \) which produces current flow in harvesting coil (Fig.2b). As a result of the interaction current and magnetic field, an electro-dynamic force (Lorentz force) \( F_{ME} \) acting on moving elements is generated. The fundamental relations describing electro dynamic force \( F_{ME} \) and electromotive force \( E_{EM} \) are described by equations:

\[
F_{ME} = \alpha \cdot i \\
E_{EM} = \alpha \cdot \dot{r}
\]  

where \( \alpha \) is coupling coefficient. The magnetic “suspension” of moving magnet is replaced by linear spring \( (k_3) \). This modification is based on the literature, where it is suggested to use an equivalent linear [2] or nonlinear [6] spring. Whereas, in the electrical part coil is modelled as a series connection of inductance \( (L_{Coil}) \) and resistance \( (R_{Coil}) \). The electrical circuit is completed by a load resistor \( (R_{Load}) \). In this paper coupling coefficient is approximated as a linear function of coil resistance \( \alpha = \alpha_0 R_{Coil} \) [8]. The parameter \( \alpha_0 \) have constant value depending on the construction of electromagnetic harvester.

![Figure 1. Scheme of an autoparametric system with an electromagnetic energy harvester](image1)

![Figure 2. Models of mechanical (a) and electrical (b) parts.](image2)
The mathematical description of the motion of the mechanical part was obtained using Lagrange’s equations of the second kind

\[
\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{q}_j} \right) - \frac{\partial L}{\partial q_j} + \frac{\partial D}{\partial \dot{q}_j} = 0, \tag{2}
\]

where the Lagrangian \( L \) is defined as a difference of kinetic \( T \) and the potential \( V \) energies, \( L = T - V \), \( D \) is Rayleigh’s dissipation function. The generalized coordinates \( q_j \) (j=1,2,3) are denoted respectively as \( x, \phi, r \).

The total kinetic energy \( T \) of the mechanical part is defined as

\[
T = \frac{1}{2} M \dot{x}^2 + \frac{1}{2} m \dot{x}^2 + \frac{1}{2} I_\phi \dot{\phi}^2 + m \phi \dot{x} x \sin \phi + \frac{1}{2} m_m \dot{\phi}^2 + \frac{1}{2} m_m \dot{r}^2 + \frac{1}{2} m_m \dot{r}^2 (R + r)^2 - m_m \dot{r} \cos \phi + m_m \dot{\phi} (R + r) \dot{x} \sin \phi \tag{3}
\]

whereas, the total potential energy \( V \) has the form

\[
V = M g x + m g (x + s - s \cos \phi) + m_m g (x + R - (R + r) \cos \phi) + \frac{1}{2} k_1 x^2 + \frac{1}{2} k_2 (x - x_0)^2 + \frac{1}{2} k_3 r^2 + F_{ME} \tag{4}
\]

Rayleigh’s dissipation function takes into account the existence of a linear damper (\( c \)) and linear air resistance (damping coefficient \( c_1 \)). This function is defined as

\[
D = \frac{1}{2} c \dot{x}^2 + \frac{1}{2} c \dot{\phi}^2 \tag{5}
\]

Applying Lagrange’s equation (2) the description of the motion are obtained in the form

\[
M \ddot{x} + m \ddot{x} + m \dot{\phi} x \sin \phi + m \phi \ddot{x} \cos \phi + m_m \ddot{\phi} \cos \phi + 2 m_m \dot{\phi} \sin \phi + m_m \ddot{\phi} \cos \phi + k x + c \dot{x} = Q \sin \omega t \tag{6}
\]

\[
I_\phi \ddot{\phi} + \frac{1}{2} \dot{m} \ddot{x} \sin \phi + m_m \ddot{\phi} \cos \phi + m_m \ddot{\phi} \cos \phi + k x + c \dot{x} = Q \sin \omega t \tag{7}
\]

\[
m_m \ddot{r} - m_m \dot{x} \cos \phi + m_m \ddot{\phi} \cos \phi + \alpha \dot{r} = 0 \tag{8}
\]

where: \( k = k_1 + k_2 \), \( Q \sin \omega t = k_3 x_0 \), \( F_{ME} = ai \) and the gravity force in equation (6) \( g(M + m + m_m) \) is not included, because this force is balanced by static preload in springs \( k_1 \) and \( k_2 \).

The electrical part is presented as a circuit, where \( E_{EM} \) is electro-motive force. The equation for this part can be written in the standard form [7]

\[
L_{C_{coil}} \ddot{i} + R_{Total} \dot{i} = \alpha \dot{r}, \tag{9}
\]

where: \( R_{Total} = R_{C_{coil}} + R_{Load} \), \( E_{EM} = ai \).

Equations (6)-(9) describe the dynamics of tested system with four degrees of freedom.

The parameters for the main object with the added pendulum are estimated from an experimental laboratory rig. In this paper we applied the following dimensional values: \( M = 0.65 \text{kg}, m = 0.265 \text{kg}, s = 0.0425 \text{m}, k = 1600 \text{N/m}, k_2 = 1100 \text{N/m}, c = 10 \text{Ns/m}, c_1 = 0.01 \text{Nm/rad}, \)
$I_0=0.000496 \text{kgm}^2$. Other parameters for the electromagnetic harvester have been adopted from the literature [1, 7]: $m_u=0.02 \text{kg}$, $R=0.0375 \text{m}$, $k_3=2000 \text{N/m}$, $L_{\text{ Coil}}=0.001 \text{H}$, $R_{\text{ Total}}=1200 \Omega$, $\alpha=3.5 \text{N/A}$ or $3.5 \text{Vs/m}$.

3 NUMERICAL RESULTS

Numerical calculations are carried out in Matlab software using $\text{ode15i}$ algorithm. This algorithm is variable order method and it solve fully implicit differential equation in general form

$$f(t, y, \dot{y}) = 0$$

where $y$ is vector of state variable. In considered system it is $y = [x, \dot{x}, \phi, \dot{\phi}, r, \dot{r}, i]$. The numerical calculations have been made for the following parameters of excitation: frequency nearby the main parametric resonance $\omega=41 \text{rad/s}$, the amplitude of kinematic excitation from 0 to 0.05m.

Figure 3 shows result of bifurcation analysis. For successive bifurcation parameter the numerical simulation always starts from the same initial condition $[x, \dot{x}, \phi, \dot{\phi}, r, \dot{r}, i]_{\text{initial}} = [0, 0, \pi / 2, 0, 0, 0]$. Presented diagrams demonstrate the ability to perform different types of motion by the pendulum. In the area, where excitation amplitude is from 0 to 0.008m the pendulum does not move. The angular velocity of the pendulum is zero (no energy harvesting). Next region (from 0.0081 to 0.0205m) is range of dynamic elimination of main object vibrations. The pendulum swings and works as a nonlinear vibration absorber performing subharmonic motion visible in the diagram by two points. For larger excitation amplitudes there are regions where the pendulum can perform chaotic motion (0.0206 -0.0344 and 0.0478-0.05) or full rotation (0.0345-0.0477).

Figures 4, 5 and 6 show the system responses for selected specific motion of the pendulum. In all presented cases it is possible to harvest energy from the pendulum motion. We can see a current flow in the electrical circuit. However, determining of the efficiency of harvested energy is difficult. The simulation conditions (for example for different amplitude of excitation) do not allow for a clear quantitative comparison of the signals $i$ from figures 4, 5, 6.
Figure 4. Times series of system responses: $x(t)$ (a), $\phi(t)$ (b), $r(t)$ (c), $i(t)$ (d) for $x_0=0.01$m (pendulum swings).

Figure 5. Times series of system responses: $x(t)$ (a), $\phi(t)$ (b), $r(t)$ (c), $i(t)$ (d) for $x_0=0.035$m (pendulum rotates).
Differential equations of motion include inertial couplings. In the presented case inertial moment of moving magnet with respect the axis of pendulum rotation (point 0 in Fig.1) is about $0.0000281 \text{kgm}^2$ (when $r=0\text{m}$). This value is close to be eighteen times smaller than the inertial moment of pendulum. Whereas, the mass of the moving magnet is about thirteen times less than the mass of the pendulum. The relations of inertia are crucial to dynamics of the pendulum in order to maintain small vibrations of moving magnet. Presented analysis completed by tests for modified model, when moving magnet has been fixed ($r, \dot{r}, \ddot{r}$ in equation (6) and (7) have been equal to zero). The absolute difference between signals $x$ and $\varphi$ from original system and modified system are very, very small. In practice responses of main object and pendulum for both systems are almost the same. The added electromagnetic harvester does not disturbed the existence of the vibration absorption phenomenon. Figure 4 shows possibility of a coexistence of two effects: nonlinear vibration reduction and energy harvesting. In further work we plan to optimize parameters of proposed harvester to allow maximization of both effects: nonlinear vibration absorption and energy harvesting.

4 CONCLUSIONS

In this paper a numerical study of application of the device for vibration absorption and energy harvesting is presented. We demonstrate that the pendulum with added electromagnetic harvester allows achieving both objectives. Numerical results for the proposed nonlinear model of tested system show possibility of energy harvesting from different kind of the pendulum motion. This research will be continued in order to take into account additional non-linearities in the harvester model.

Acknowledgments This work was financially supported under project of National Science Centre according to decision No. DEC-2013/11/D/ST8/03311.
REFERENCES


A parametric approach to the generation of multidisperse granular flows for particle simulations

Alessandro Tasora*, Ida Critelli#, Hammad Mazhar∇

* Department of Industrial Engineering
Università degli Studi di Parma
Parco Area delle Scienze 181/A, 43100, Parma, Italy
tasora@ied.unipr.it

# Department of Mechanical Engineering
Politecnico di Milano
Via La Masa 1, 20156, Milan, Italy
ida.critelli@polimi.it

∇ Department of Mechanical Engineering
University of Wisconsin
Madison 2035ME, 1513 University Avenue Madison, WI 53706-1572
hmazhar@wisc.edu

ABSTRACT

In this paper we present an innovative approach to the systematic generation of multi-disperse granular flows for particle simulations in multibody problems. Classical approaches to this problem are based on probability distributions for particle sizes, such as granulometric curves, whereas in our formulation particles can be sorted from a probability space that involves also other parameters such as density, aspect ratio, geometric properties, and so on. The method is fully parametric, in the sense that most properties of the generated particles can be assigned to statistical distributions, using a modular implementation in our simulation software. Probability distributions are introduced as C++ objects; we implemented a library of distributions of various types, either as probability density functions or cumulative distribution functions.

Keywords: Multibody, granular flows, particle simulation, probability, distribution.

1 INTRODUCTION

This paper presents a novel approach to the generation of particle flows where the properties of particles are defined by configurable probability distributions.

Usual methods for the initialization of granular material in multi-body simulations are based on the generation of particles in random positions, where the size of the particles follows a statistical distribution given by a single granulometric curve. Although this is sufficient in various applications, there are cases where other properties of the particles should be subject to randomization, for example friction coefficients or material density.

To this end we developed a new systematic approach where particles can be sorted from a configurable hierarchical probability space that involves also other parameters such as density, aspect ratio, geometric properties, and so on. When possible, the random variates are generated using the Smirnov theorem on inverse transform sampling.

Shapes of the particles can belong to different classes, for instance convex hulls, spheres, beams, and so on; those classes are unlimited since they can be extended by means of object oriented programming, and each class introduces statistical distributions about its geometric parameters. Each shape generator can be assigned to a particle family; those families are sorted according to a discrete statistical distribution. Particle families can be organized in hierarchical trees of sub-families, hence obtaining a high level of configuration of the system.

Figure 1 shows an example of a random generation of particles in a multidisperse granular flow from a single particle class, namely convex hulls that represent fragments in a shredding process.
Here shape parameters follow statistical distributions about size, aspect ratio and amount of detail, to name a few. As shown in Figure 2, multiple families with particles of different classes can be mixed in a stochastic way, according to a provided discrete statistical distribution.

We tested this framework within our multibody simulation software whose formulation, based on Differential Variational Inclusions (DVI), can target problems with a massive number of particles with frictional contacts [1]. To this end we performed benchmarks with hundreds of thousands of particles on a parallel computing architecture [2].

This approach to the generation of multidisperse granular flow originated from our researches on separation processes used in recycling plants of electronic waste [3], where particles have stochastic properties of geometry, density, electric charge and material type; then we extended this concept to other applications that benefit from this parametric framework, for instance in simulations of earth-moving machines interacting with granular soil.

2 PROBABILITY

In this section we will recall some classical terms and propositions related to the concept of probability [4],[5]. These definitions will be used in the rest of the section when we will discuss the algorithms that we developed for generating the variates.

We introduce a probability space

\[(\Omega, \mathcal{F}, P)\]

based on a sample space \(\Omega\), on a \(\sigma\)-algebra \(\mathcal{F}\) and on a probability measure \(P\).

- The sample space \(\Omega\) is a non-empty set containing samples \(\omega \in \Omega\).
- The \(\sigma\)-algebra \(\mathcal{F}\) is a collection of subsets of \(\Omega\), its elements \(A\) are called events: \(A \in \mathcal{F}\), so \(A \subset \Omega\). Then, \(\mathcal{F} \subset 2^\Omega\), and in many cases \(\mathcal{F} = 2^\Omega\), the power set of \(\omega\). The \(\sigma\)-algebra \(\mathcal{F}\) is closed under finitely-many operations of complement, union and intersection.
- The probability measure \(P\) is a function \(P : \mathcal{F} \rightarrow [0,1]\). In case of discrete events \(\omega_i\), each with a discrete measure \(p(\omega_i)\), it holds \(P(A) = \sum_{\omega_i \in A} p(\omega_i)\). Otherwise it is \(P(A) = \int_A dP\).

The Kolmogorov theory of probability requires the following axioms:

- \(P(A) \in \mathbb{R}^+\) for \(A \in \mathcal{F}\),
- \(P(\Omega) = 1\),
- \(P(A_1 \cup A_2 \ldots \cup A_n) = \sum_i P(A_i)\) for disjoint sets.
As a simple example, \( \Omega \) can be an Euclidean space \( \mathbb{R}^n \) or a Lebesgue measurable subset of it, so \( \mathcal{F} \) will be the \( \sigma \)-algebra of all Lebesgue measurable subsets of \( \Omega \), and \( P \) will be a Lebesgue measure. Other useful propositions are:

- **A random variable** is a \( (\mathcal{F}, \mathcal{B}) \)-measurable function \( X : \Omega \to B \) where \( (B, \mathcal{B}) \) is a measurable space. In many cases it is simply a scalar such as \( X : \Omega \to \mathbb{R} \).

- Given measurable spaces \( (X_1, \mathcal{F}_1) \) and \( (X_2, \mathcal{F}_2) \), a \( (\mathcal{F}_1, \mathcal{F}_2) \)-measurable mapping \( f : X_1 \to X_2 \) and a measure \( \mu : \mathcal{F}_1 \to [0, +\infty) \), the pushforward measure, also called *distribution*, is defined as the measure \( f_* (\mu) : \mathcal{F}_2 \to [0, +\infty) \) expressed as
  \[
  [f_* (\mu)](S) = \mu (f^{-1} (S))
  \]
  for \( S \in X_2 \).

- The *probability of a random variable* in the discrete case is:
  \[
  P[X = s] = P(X^{-1}(s)) = [X_* (P)](s) = f_X (s)
  \]

- The *probability of a random variable* in the continuum case, that is the probability that \( X \) falls in some interval \( S \), is:
  \[
  P[X \in S] = P(X^{-1}(S)) = [X_* (P)](S)
  \]

- The *probability mass function*, for a discrete random variable with a counting measure \( d\nu \), is defined as \( f_{PMF} \) in:
  \[
  P[X = s] = \int_{\{s\}} f_{PMF} d\nu
  \]

- The *probability density function*, for a continuous random variable with a reference measure \( d\mu \) on \( (B, \mathcal{B}) \), is defined as \( f_{PDF} \) in:
  \[
  P[X \in S] = \int_S f_{PDF} d\mu.
  \]

- The \( f_{PDF} \) is a Radon-Nykodym derivative:
  \[
  f_{PDF} = \frac{d[X_* (P)]}{d\mu}.
  \]

- A random variate \( X_i \) is a computer-generated outcome of a random variable sorted from its \( (B, \mathcal{B}) \) space, according to specified probability density functions.

The goal of this work is to generate particles whose properties are random variates. From the point of view of the algorithmic implementation, regardless of the underlying \( (\Omega, \mathcal{F}, P) \), one can see that the required information is the \( (B, \mathcal{B}) \) space and the \( f_{PDF} \) (or the \( f_{PMF} \)) defined on it.

### 2.1 Absolutely continuous univariate distributions

Among the simplest cases there are the absolutely continuous univariate distributions, where \( B = \mathbb{R}, X \in \mathbb{R} \) and \( \mu \) is the Lebesgue measure. In this case:

\[
P[a \leq X \leq b] = \int_a^b f_{PDF} (x) \, dx
\]
In this case it is often used the cumulative distribution function, that is defined as

\[ F_{CDF}(x) = \int_{-\infty}^{b} f_{PDF}(t)dt \]

For example, a case that belongs to this class is the problem of generating particles that only differ by a diameter \(d\), where \(f_{PDF}(d)\) will be given by some granulometric curve obtained experimentally. Sometimes, in granular flow applications, diameter distributions are provided as \(F_{CDF}(d)\), but this does not constitute a problem as it is always possible to compute \(f_{PDF}\) from \(F_{CDF}(d)\) and viceversa. For instance, \(f_{PDF}(d)\) distribution could be a Weibull distribution:

\[ f_{PDF}(d) = \frac{k}{\lambda} \left( \frac{d}{\lambda} \right)^{k-1} e^{-\left( \frac{d}{\lambda} \right)^k}, \quad \text{for } x \geq 0. \]

In some cases the variable space \(B\) could be countable. We say that this case corresponds to discrete random variables.

Suppose, for example, that one needs to define a distribution of particles that differ only for two material types: in such case, \((B, \mathcal{B})\) could contain just two samples as \(B = \{B_1, B_2\} = \{\text{copper, steel}\}\), with \(f_X(\omega_1)\) and \(f_X(\omega_2)\) two measures called probability masses.

### 2.2 Multivariate distributions

A more advanced case is represented by (continuous) multivariate distributions, where \(B = \mathbb{R}^n\) and \(X = \{X_1, X_2, ..., X_n\} \in \mathbb{R}^n\). The \(f_{PDF}(x_1, x_2, ..., x_n)\) is a generic \(\mathbb{R}^n \rightarrow \mathbb{R}^+\) function that must satisfy the Kolmogorov axiom, with

\[ \int_{x_1} \int_{x_2} \cdots \int_{x_n} f_{PDF}(x_1, x_2, ..., x_n)dx_1dx_2...dx_n = 1 \]

A special case is when all variables are independent, that leads to:

\[ f_{PDF}(x_1, x_2, ..., x_n) = f_{PDF_{x_1}}(x_1)f_{PDF_{x_2}}(x_2)...f_{PDF_{x_n}}(x_n) \]

that is, \(\Pi^n f_{PDF_{x_i}}(x_i)\) with \(n\) marginal distributions \(f_{PDF_{x_i}}(x_i) : \mathbb{R} \rightarrow \mathbb{R}^+\).

These are just like univariate distributions seen above, so in this case a multivariate distribution can be built as a collection of univariate distributions. Note that this can be extended also to discrete distributions.

### 3 RANDOM VARIATES: THE ALGORITHM

There are various approaches to the computer generation of random variates given a \(f_{PDF}\). We use the Smirnov transform.

We assume that the computer has a way to generate a random scalar variate \(U_i\), a random variable \(U\) with uniform probability density in \([0, 1]\). This is achieved using pseudo-random number generators as the Mersenne twister, or more sophisticated algorithms. From the programming point of view this boils down to a single function evaluation such as \(U = \text{rand}()\).

The process for computing a random variate \(Y_i\) from a \(f_{PDF_i}\) is the following:

- The \(F_{CDF_i}\) is computed from \(f_{PDF_i}\), either because it is known analytically (ex. with Weibull distribution) or because it is obtained by numerical integration (ex. when \(f_{PDF_i}\) is provided with sampled points or a spline).

- The inverse of \(F_{CDF_i}\), that is \(F_{CDF_i}^{-1} : [0, 1] \rightarrow \mathbb{R}\), is obtained, either analytically or in form of a polygonal function with numerical approximation.

- Each time \(Y_i\) is needed, one evaluates \(Y_i = F_{CDF_i}^{-1}(U_i)\). That is:

\[ Y_i = F_{CDF_i}^{-1}(\text{rand}()). \]

This is explained in Fig.3.
The case of independent variables simplifies a lot the problem of variates: for example particles could be generated with generating a variate from continuous Weibull distribution for sizes, then a second variate could be generated on a discrete distribution for determining the material, ex. steel or plastic.

However, consider a more advanced case where one might want to randomize, for instance, the particle electric charge: this means that there would be a third distribution for the electric charge; however this is not independent from the material, and maybe also from the particle size. In this case, Eq.1 does not hold, and \( f_{PDF} (x_1, x_2, \ldots, x_n) \) should be provided as a monolithic function. This is not easy, especially if one want to match statistical data that comes from experiments.

In the rest of this paper we describe a method that allows the definition of \( f_{PDF} \) in a versatile way, when \( (B, \mathcal{B}) \) has a tree-like hierarchical structure. This encompasses also the case of independent continuous multivariate distributions but also enables the definition of some types of more generic distributions with dependent variables, at least at a discrete level.

4 MODULAR, OBJECT-ORIENTED IMPLEMENTATION OF VARIATES

In the following will discuss how to define a non trivial space \( (B, \mathcal{B}) \) from the programming point of view, and how to generate random variates \( X \) from it.

We assume that the variate \( X_i \) must contain all the needed information for building a particle that must be added in a granular flow.

This development follows the observations that, in many applications involving granular matter, one has statistical and experimental data in this form:

- there are \( n \) types of materials mixed together,
- for each material one has a percentual (in terms of particle number, or mass or volume) respect to the total, that is \( n \) discrete measures in a \( f_{PMF} \),
- for each material type, particles have different material properties, and these material properties might be in a different number, for example spheres have radius as a property, cylinders have height and diameter, etc. So the cardinality of \( X_i \) is not constant and might change with \( i \), and this leads to different families of particles.
- within a single family of particles one has statistical data in form of univariate distributions.

As an example, consider a granular material used in a plant for processing waste flows.\(^1\) Assume

\(^1\)Experimental data for this example can be obtained experimentally with simple processes, such as using sieves, magnets, etc.
one has a 70% of plastic and a 30% of steel. Within the steel particles, one might have, for instance, 80% of chopped electrical wires, made with copper, and 20% of shred steel plates. In the former case, particles are shaped as bent wires and one could introduce a continuous probability distribution about the length and the curvature, whereas in the second case the splinters could be random polytopes that follow a continuous distribution in terms of chordal size and size ratios. These distributions could be determined using microscopes and computer vision tools. And so on for further sub-families of particles.

In the end, one can see that many cases fall in a scheme: there is a hierarchical tree of particle families, each with distinct properties, and there are (uniform or discrete) distributions assigned to the leafs of that tree. Each tree level has a discrete probability. See Fig.4 for an example.

![Figure 4](https://example.com/figure4.png)

**Figure 4.** Example of tree-like configuration of the variate generator.

From the programming point of view this translates into the following concepts.

- Discrete and continuous monovariate distributions are C++ objects inherited from the base class `ChDistribution` that exposes the interface for generating a random variate $X_i \in \mathbb{R}$. We implemented a few distribution types:
  - Weibull
  - Zhang
  - Normal
  - Uniform in a $[\text{min}, \text{max}]$ interval
  - Sampled continuous, with $f_{PDF}$ passing through user points
  - Sampled discrete, with $f_{PMF}$ defined by user points
  - etc. (other can be implemented through inheritance)

- The random variate $X_i$ for the granular matter generation is a C++ object: the particle to be simulated. Particle objects inherit the base class `ChBody` for moving objects in space, with some shape and material.

- In our software there are many particle types, and thank to C++ inheritance, one can always extend the property types and properties using sub-classes of the root class `ChBody`

- A random variate $X_i$ (i.e the random particle) is generated by a `ChRandomShapeCreator`-inherited class.
• We implemented a few particle generators inherited from ChRandomShapeCreator. For instance we have
  - ChRandomShapeCreatorSphere,
  - ChRandomShapeCreatorConvexHull,
  - ChRandomShapeCreatorShavings,
  - ChRandomShapeCreatorBoxes,
  - etc.

• Objects inherited from the ChRandomShapeCreator class can contain various amounts of ChDistribution objects. For example the ChRandomShapeCreator contains a ChDistribution for generating a variate for the sphere radius.

• The \((B, \mathcal{B})\) space has a tree-like structure.

• At the root of the tree is a C++ object that we call ChRandomShapeCreatorFromFamilies, a C++ object that contains finite collection of other ChRandomShapeCreator objects, each with a discrete probability, so the container itself embeds a monovariate discrete distribution where the point masses are the sub-creator percentages.

• A ChRandomShapeCreatorFromFamilies can contain another object of the same class, hence leading to trees with more than one level.

• A tree leaf is an object of ChRandomShapeCreator class.

This means that, when a variate is needed, the root of the tree is asked to generate a particle. If the root is a ChRandomShapeCreatorFromFamilies, it will perform a discrete variate to pick the proper sub-creator (according to their specified percentual values\(^2\)), and then it will delegate the particle creation to that sub-creator.

This process proceeds recursively until the leaf is found.

Note that this process, that builds a multivariate particle \(X_n\), boils down to a sequence of monovariate random variables, as happens in models with independent variables. Nonetheless, due to the tree-like nature of this process, the result is more generic than the case of independent probability, and define one special type of conditional probability that encompasses a large number of practical cases.

5 OPTIMIZATIONS

In order to improve the efficiency of the collision detection algorithm we designed an innovative optimization strategy where the probability space is discretized by pre-sampling it with a limited number of particle that we call particle prototypes.

At the cost of introducing a quantization in continuous distributions, this optimization has the benefit that all following particles are just clones of the original prototypes. This saves memory because geometric details of clones can be referenced by sharing a single data structure.

For instance, say the original variate generator should create a million of convex hulls with diameters whose probability follows a smooth Weibull distribution. Before the simulation starts, a distribution quantizer creates a limited number of prototypes, say 500, and store them in an array. Later, when particles must be generated during the simulation, the required particles are cloned

\(^2\)We remark that if the percentage is expressed in terms of number of particles, this amounts to using the Smirnov transform as expressed in Eq.2, but if the percentage is in terms of volume or mass, this requires some additional steps that are not discussed here for brevity.
from the prototypes of that array, sampled with uniform distribution. Since in our software the
geometric data (the vertexes of the convex hull in this case) can be shared among clones, in this
example one would achieve a 2000x factor in memory-saving.
This optimization requires that the simulation software implements shared pointers and object
cloning, or at least serialization methods - things that are well understood in professional C++
development.
Less memory footprint also means faster code, because modern computing architectures are mostly
data-starving, and out-of-cache accesses affect negatively the performance.

6 PARTICLE EMITTERS AND PROCESSORS
Suppose a particle has been created as a variate from the \((B,\mathbb{R})\) space, thank to the tree-like
hierarchical algorithm discussed in the previous section. The following problem is to place it in
space and to assign an initial velocity.

We designed an object-oriented approach to this problem as well.
In fact there is a class of objects called ChEmitter that is in charge of calling the particle creator as
many times as necessary to satisfy a particle/second or a mass/second value, hence creating many
particles per each timestep, automatically. This simulates the effect of an outlet that generates a
continuous flow of particles.

Not only the ChEmitter contains a ChParticleCreator object with the tree that defines the
particle properties, but it also contains other C++ objects of stochastic nature. In detail, it contains:

- a randomizer object that defines the random initial shape and properties of the particle, that
  is the already discussed ChParticleCreator
- a randomizer object that defines a random initial position for the particle; we implemented
  some ready-to-use randomizer such as:
  - point outlet,
  - rectangular outlet,
  - circular outlet,
  - random point on a parametric line \((x = f(u), \text{ and } u \text{ too can have a non uniform probability density assigned through a ChDistribution object})\),
  - random point on a biparametric surface \((x = f(u,v), \text{ with non uniform density in } u \text{ and } v \text{ if needed})\),
  - random point on a triparametric volume \((x = f(u,v,w))\),
  - etc. (other can be implemented via C++ inheritance).
- a randomizer object that defines a random initial rotation for the particle; we have some
  ready-to-use randomizer such as:
  - constant alignment,
  - randomized rotation with uniform sampling of \(\text{SO}(\mathbb{R},3)\),
  - etc.
- a randomizer object that defines a random initial speed for the particle; among these avail-
  able by default:
  - constant initial speed,
  - constant direction, stochastic modulus (defined via a ChDistribution object),
– stochastic direction, stochastic modulus,
– etc.

In Figure 5 one can see an example of use of the emitter object. Note that the emitter can be moved during the animation and its flow rate can be changed.

Figure 5. Example of particle emission from one rectangular outlet and two point outlets.

These emitters can be used to generate granular flows during the simulation, however in many cases one needs a method to generate all particles at once, for instance when one has to generate a granular soil for simulating machine-earth interaction. In such a case, we designed an object that has some commonalities with emitter objects: we defined a class of filler objects because they can fill 3D volumes with packed particles.

A ChParticleFiller contains:

• a ChParticleCreator randomizer that defines the random initial shape and properties of the particle,
• a volume sampler for the initial position of the particle; some are available by default:
  – uniform grid,
  – uniform hexahedral lattice,
  – Poisson 2D sampling,
  – Poisson 3D sampling

Note that the Poisson 3D sampling is able to achieve low void ratio packings without much overlapping between particles, so it is often used to initialize soils and dense packings of particles, however this beneficial property is valid only when particles are spherical. There is no easy way to generate a close packing of generic shapes, so when non-spherical particles are used, one should consider generating an initial loose packing (possibly also reverting to using an emitter that generate a fall of particles) and then letting the gravity or some other vertical force field do the packing of the particles in a pre-processing stage. Once particles are packed, their state can be serialized to disk so that the next time the preprocessing stage can be skipped.
7 PARTICLE FLOW DNA

The way that particle families, generators and statistical distributions are assembled can be defined by an optional configuration file that leverages the JSON serialization format; we refer to this information as the DNA of the granular flow.

Since the \((B, \mathcal{B})\) space and its \(f_{PDF}\) is defined as a hierarchy of C++ objects, one can use a serialization schema to transform the data structure into a JSON archive (one could also use other formats, here, for instance XML). Viceversa one can write a JSON file with an ASCII editor, and load it into the simulator: the deserialization will automatically recreate the transient data structure. This requires that the serialization mechanism is able to cope with shared pointers, polymorphic objects, class factories and other advanced concepts in C++ programming.

An example of JSON file that defines a simple granular flow DNA is reported in the following.

```json
"emitter" : {
  "outlet_height" : 0.1,
  "outlet_width" : 0.182,
  "particles_per_second" : 15000,
  "use_particle_reservoir" : true,
  "particle_reservoir" : 30000,
  "particle_creator" : {
    "type" : "ChRandomShapeCreatorFromFamilies",
    "families" : [{
      "type" : "ChRandomShapeCreatorSpheres",
      "diameter_distribution" : {
        "type" : "ChMinMaxDistribution",
        "min" : 0.003,
        "max" : 0.003
      },
      "density" : {
        "type" : "ChConstantDistribution",
        "value" : 8400
      },
      "add_collision_shape" : true,
      "add_visualization_asset" : true,
      "material_type" : "metal",
      "color" : [0.5, 0.5, 0.3],
      "probability" : 0.3
    }, {
      "type" : "ChRandomShapeCreatorBoxes",
      "Xsize_distribution" : {
        "type" : "ChMinMaxDistribution",
        "min" : 0.002,
        "max" : 0.003
      },
      "sizeratioZ" : {
        "type" : "ChMinMaxDistribution",
        "min" : 0.5,
        "max" : 1.0
      },
      "density" : {
        "type" : "ChConstantDistribution",
        "value" : 946
      },
      "add_collision_shape" : true,
      "add_visualization_asset" : true,
      "material_type" : "plastic",
      "color" : [0.3, 0.6, 0.6],
      "probability" : 0.7
    }
  }
}
```

This opens the road to parametric studies where such granular flow DNA is automatically changed during multiple simulation runs, for instance in optimization or in sensitivity analysis.

8 EXAMPLES

In Figures 6, 7, 8 and we show some example of application of the discussed methodology. Note that particles of different classes can be mixed.

9 CONCLUSIONS

We presented an approach to the systematic generation of multi-disperse granular flows for particle simulations in multibody problems.
Figure 6. Monodisperse granular flow with a sphere generator. A Zhang distribution is applied to diameters.

Figure 7. Monodisperse granular flow with a splinter generator. A Zhang distribution is applied to splinter chord, and a uniform distribution is used for the aspect ratio and number of vertexes.
Figure 8. Monodisperse granular flow with a generator that creates bent beams. Various continuous random distributions are used for geometric properties and for surface color.

Figure 9. Monodisperse granular flow with a generator that creates boxes. Continuous distributions have been used for size ratios, whereas a discrete distribution has been used to select between three surface coatings, represented as three distinct colors.
An original method for the stochastic generation of particles with varying shapes and properties has been designed in form of a hierarchical tree-like structure of C++ objects. Those objects introduce stochastic parameters and can be extended with inheritance and polymorphism. The framework is completed with a JSON serialization scheme to import/export the DNA of the granular flow, and with a modular system for emitting particles from virtual outlets or volumes. This approach has been successfully tested in real applications, involving the simulation of waste processing devices.

REFERENCES


Coupled Analysis of Thermo-Fluid-Flexible Multi-body Dynamics for a Two-Dimensional Engine Nozzle

Wonjong Eun*, Jaewon Kim#, Chanhoon Jung*, SangJoon Shin*, Oh-Joon Kwon#, and Olivier A. Bauchau†

* Department of Mechanical and Aerospace Engineering
Seoul National University
1 Gwanak-ro, Gwanak-gu, Seoul, 157-744, Republic of Korea
[eun0914, ssjoon, chan3142]@snu.ac.kr

# Department of Aerospace Engineering
Korea Advanced Institute of Science and Technology
291 Daehak-ro, Yuseong-gu, Daejeon, Republic of Korea
[guonetoo, ojkwon]@kaist.ac.kr

† Department of Mechanical and Aerospace Engineering
Hong Kong University of Science and Technology
Clear Water Bay, Kowloon, Hong Kong
bauchau@ust.hk

ABSTRACT

Various components in an engine nozzle are modeled as flexible multi-body components which are operated under high temperature and pressure. In this paper, in order to predict complex behavior of an engine nozzle, thermo-fluid-flexible multi-body dynamic coupled analysis framework was developed. Temperature and pressure on the nozzle wall were obtained by the CFD flow analysis for two-dimensional nozzle. The pressure and material properties according to temperature were delivered to DYMORE. Then nonlinear multi-body dynamics analysis was performed. Heat conduction and thermal analyses were conducted by MSC.NASTRAN. The present framework was validated for a simple nozzle configuration by using an one-way coupled analysis. Both-way coupled analysis was also performed for the simple nozzle with an imaginary joint clearance, and an asymmetric flow was observed. Finally, total strain result of a realistic nozzle configuration was obtained by the one-way and both-way coupled analyses.

Keywords: Multi-body dynamics, Fluid-structural coupled analysis, Coupled analysis framework.

1 INTRODUCTION

Thrust vector control is required in aircrafts, rockets, or other vehicles to adjust the thrust direction from its engine and vary the attitude or angular velocity. For thrust vector control, a jet engine exhaust nozzle consists of numerous thin plates and joints to change its flow path and flux area. To perform these tasks, the engine nozzle consists of a number of movable thin plates and joints, as shown as Figure 1. The nozzle can change and maintain its configuration by actuators. Inside the nozzle, high temperature and pressure flow impinge on its surface. It may induce nonlinear complicated stress and strain patterns in the structural components.

Because of various components of an engine nozzle, it is modeled as a flexible multi-body structure. Computational flow analysis is essential to calculate the pressure and temperature. Also, the changes of material properties according to high temperature should be considered. To predict the complex behavior of an engine nozzle under such high temperature and pressure, thermo-fluid-flexible multi-body dynamic coupled analysis will be generally required.

In the previous literature, few attempts to analyze an engine nozzle have been reported. Whitaker and Gowadiat [1] conducted conceptual study of aircraft thrust vectoring using flexible two-dimensional nozzle configuration. Schwane and Xia [2] performed fluid-structure interaction stability analysis for three-dimensional nozzle. Their numerical results were validated with hypersonic wind tunnel experimental results. Murugappan et al. [3] conducted fluid-structure coupled analysis for a flexible nozzle. They employed fully closed nozzle...
configuration. Wang et al. [4] introduced tightly coupled algorithm to simulate the nozzle deformation. They performed transient aeroelastic nozzle start-up analysis at sea level. However, multi-body configuration and material properties variation were not implemented in those investigations.

Recently, multi-body dynamic analysis programs, such as ADAMS [5], DADS [6] and SAMCEF-Mecano [7], have been developed. However, these programs provide only rigid structural elements and joints. They show limitation to analyze an elastic deflection on engine nozzle components under high temperature and pressure. A flexible multi-body dynamic analysis program, RecurDyn [8], provides elastic bodies and various joints. An arbitrary complex modeling is possible by assembling these components. However, to apply distributed loads on its beam element, nodal forces need to be specified manually.

In this paper, thermo-fluid-flexible multi-body dynamic coupled analysis for two-dimensional nozzle was attempted. Nonlinear flexible multi-body dynamic analysis was conducted by DYMORE [9]. In-house CFD program [10] was developed with mixed unstructured grid for flow analysis. Thermal analysis was also performed with MSC.NASTRAN [11]. And coupled analysis procedure was established among these analyses. For the coupled analysis, simple and realistic two-dimensional nozzle configurations were selected. Then temperature and pressure on the nozzle wall were predicted by the steady-state flow analysis. The temperature results were delivered to NASTRAN and thermal analysis was conducted. Nonlinear flexible multi-body dynamics analysis was performed by DYMORE. The present interface module allowed the internal result exchanges. By adopting the present analytical procedure, prediction on the complicated behavior of a jet engine nozzle will be enabled.

2 THEORETICAL ELEMENTS OF THE PRESENT ANALYSIS

2.1 Multi-body dynamic analysis, DYMORE

DYMORE is a finite element based tool for the analysis of nonlinear elastic multi-body systems [9]. It allows the modeling of arbitrary complex configurations through the assembly of deformable bodies and numerous joint elements. Figure 2 depicts the six lower pairs of joints which are provided by DYMORE [9].

Deformable bodies were modeled by the finite element method, without resorting to modal approximations. In this paper, two-dimensional nozzle components were modeled as beam elements. In DYMORE, the geometrically exact beam formulation for nonlinear elastic beam is employed. Figure 3 shows the coordinates between the reference and deformed configuration of the beam. A detailed formulation of the equations of motion for beams and their finite element implementation is found in Ref. 8.

The clearance element provided in DYMORE describes the parameters associated with the definition of a clearance inside revolute joints. Consider the revolute joint depicted in Figure 4. The outer and inner races of the revolute joint with clearance are modeled as bodies K and L of
the revolute joint with radii \( \rho^k \) and \( \rho^l \), respectively. The relative distance between the candidate contact points \( z^k \) and \( z^l \) is denoted by \( q \). Such element is used in conjunction with contact, lubrication, and backlash characteristics.

By applying extreme difference or clearance condition to upper and lower joints of the present two-dimensional configuration, asymmetric structural deflection and flow result will be induced.

### 2.2 Flow analysis

In the present paper, a mixed unstructured mesh flow solver has been developed to simulate a two-dimensional nozzle. Navier-Stokes equations can be written in an integral form for an arbitrary computational domain \( V \) with boundary \( \partial V \) as

\[
\frac{\partial}{\partial t} \int_V Q\,dV + \int_{\partial V} F(Q) \cdot \vec{n}\,dS = \int_{\partial V} G(Q) \cdot \vec{n}\,dS + \int_V S(Q)\,dV
\]  

(1)

where \( Q \) is the vector of the conservative variables, and \( F(Q) \) and \( G(Q) \) denote the inviscid and viscous fluxes, respectively. The governing equations were discretized by using a vertex-centered finite volume method on unstructured grid. The flow domain was divided into a finite number of control volumes surrounding each vertex, which were made of non-overlapping
median-dual cells whose boundary surfaces were defined by the cell centroid, face centroid and mid-point of the edge of the unstructured mesh. The inviscid flux terms were approximated using Roe’s flux difference splitting scheme [12], whereas the viscous flux terms were evaluated by a modified central difference method [13]. The flow variables at each dual face were assessed by applying a least-square reconstruction technique to achieve the second-order spatial accuracy. The integration in time of the discretized equations proceeded by using a point Gauss-Seidel iteration method. To enhance the convergence speed, a local time stepping procedure was taken. To estimate the turbulent eddy viscosity, the Spalart-Allmaras turbulence model [14] was adopted.

Figure 5 shows the computational meshes and boundary conditions used in the present study. Stagnation conditions were applied to the left face of the inflow upstream of the nozzle. At the far-field boundary, the characteristic inflow/outflow boundary condition with the Riemann invariants was used. All solid walls were applied as no-slip and adiabatic boundary conditions. Exit flow boundary condition was applied on the right face that changed between constant pressure outflow and supersonic outflow depending on the local Mach number.

2.3 Thermal analysis

In this paper, material properties variation according to the temperature distribution and thermal expansion were considered. The temperature on the nozzle wall was predicted by steady-state flow analysis. In order to obtain the temperature of the nozzle components between inner and outer nozzle wall, heat conduction analysis was employed. The entire nozzle temperature distribution was predicted and applied to material properties calculation and thermal strain analysis. Heat conduction and thermal strain analyses were conducted by MSC.NASTRAN [11].

However, the present thermal analysis procedure did not consider rotational motion with respect to the revolute joints. The restriction of rotation may induce excessive strain result. Also, heat convection and radiation analyses were not included in the present prediction. In a real situation, temperature at the solid surface is reduced by film cooling effect. The present thermal analysis may thus over-predict the temperature distribution. To predict a more precise temperature distribution, heat convection and radiation effect will be included in the future.

3 COMBINED FRAMEWORK

3.1 Thermo-Fluid-Flexible multi-body dynamic coupled framework

The thermo-fluid-flexible multi-body dynamic coupled analysis framework for a jet engine nozzle was established. Figure 6 shows the flowchart for the present analysis. The five important features of the present coupled analysis framework are as follows.

- Interface between external design software and analysis

In this procedure, two-dimensional nozzle configurations were modeled by computer-aided design software, CATIA. For flow and structural analyses with nozzle configurations, external geometry information was transformed as internal data for analysis. Whereas the CATIA file
Figure 6. Flowchart of the thermo-fluid-flexible multi-body coupled analysis.

Figure 7. Conversion of pressure into distributed load.

provided fundamental geometries only, the coupled analysis required additional information, such as types of joints or boundary conditions, in order to specify the multi-body elements. The present interface module was capable of creating DYMORE input data including geometrical data from external design software and multi-body element information.

- Fluid-Flexible multi-body dynamically coupled analysis

For the first attempt, steady-state nozzle flow analysis was performed for the initial nozzle configuration. Then the pressure and temperature on the nozzle wall was obtained. The interface module received nozzle wall pressure and geometry. The nozzle wall pressure was converted in a form of the distributed loads. The linear interpolation scheme was implemented for this process (Figure 7). The distributed loads were delivered to DYMORE. Then the nonlinear flexible multi-body dynamic analysis was performed.

- Material properties variation

An exhaust nozzle is usually exposed to high temperature. Thus, changes of the material properties are induced. The flow analysis calculated the temperature on the nozzle wall. The
nozzle wall temperature was delivered to NASTRAN. In order to obtain the temperature distribution for the entire nozzle configuration, heat conduction analysis was performed. The present interface module predicted the material properties depending on the temperature results from the heat conduction analysis. MIL-HDBK-5H [15] was used in order to predict material properties variation. Figure 8 presents the relationship between temperature and material properties of the Titanium alloy, Ti-6Al-2Nb-1Ta-1Mo.

- Total strain estimation

The interface module generates DYMORE input file while considering the distributed loads and the material properties. After nonlinear structural analysis was performed in DYMORE, the structural variables, which included deformations and mechanical strain for the nozzle components, were obtained. Thermal strain analysis was also conducted with the temperature distribution from the heat conduction analysis. Finally, total strain result was obtained as a summation of the mechanical and thermal strain.

- Both-way coupled analysis

The one-way coupled analysis included fluid-structure interaction, material properties variation, and thermal analysis. However, the deflection of the nozzle configuration was not considered. Both-way coupled analysis took the deformed configuration into account for flow analysis. The deflection of the nozzle wall, which was predicted by DYMORE, was delivered back to flow analysis. Then flow analysis was conducted with the deformed nozzle configuration. Such intrinsic result exchanges were conducted for several times. By this way, loosely-coupled analytical procedure was established.

4 VALIDATIONS OF THE RESPECTIVE ELEMENT IN THE PRESENT ANALYSIS

4.1 Structural analysis validation
A simple cantilevered beam analysis was performed to validate the present structural analysis. Comparison result of transverse deflection, which was predicted by the present analysis, RecurDyn, and Euler-Bernoulli beam theory with the cantilevered beam under distributed load, is shown in Figure 9. The average difference of deflection between the present analysis and the Euler-Bernoulli beam theory was less than 1%. Thus, the validity of the present structural analysis was ascertained.

### 4.2 Flow analysis validation

The flow analysis validations were made by comparing the present flow analysis results with available experimental results [16]. Steady-state flow analyses were conducted with experimental configuration [16] for three different nozzle pressure ratio, 2.412, 5.423, and 8.780. Table 1 shows steady-state flow analyses conditions.

<table>
<thead>
<tr>
<th>Test gas</th>
<th>Ideal Air</th>
<th>Expansion ratio</th>
<th>1.797</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stagnation temperature</td>
<td>294 K</td>
<td>Reynolds number</td>
<td>$3.2 \times 10^6$</td>
</tr>
<tr>
<td>Ambient pressure</td>
<td>101.33 kPa</td>
<td>Nozzle pressure ratio (NPR)</td>
<td>2.412, 5.423, 8.780</td>
</tr>
</tbody>
</table>

Figure 10 illustrates the comparison for the pressure distribution, predicted at the nozzle wall against those measured by experiment. It reveals that the present predictions were accurate and showed good agreement with experiment, within a discrepancy of 3%. Thus, the present in-house CFD analysis was validated.

![Figure 9. Transverse deflection of cantilevered beam under distributed load.](image1)

![Figure 10. Pressure distribution at the nozzle wall.](image2)
5 NUMERICAL RESULTS

5.1 Analysis configurations

In the present paper, two different nozzle configurations were selected for the five different analysis configurations. A realistic nozzle has gaps between various elements. However, in order to implement flow analysis, nozzle surface has to be smooth without any gap. Thus, a simplified two-dimensional nozzle configuration was devised. The simple nozzle one-way coupled analysis was conducted to validate the fluid-structural coupled analysis procedure. Figure 11 shows the simple nozzle configuration.

In order to observe asymmetric flow, both-way coupled analysis of simple nozzle configuration with arbitrary joint clearance was performed. Figure 12 shows the modified configuration for the present engine nozzle including the clearance. Two joints in the upper half component were replaced by the one including a certain amount of clearance as shown in Figure 12. It may induce the pressure difference between the upper and lower wall, asymmetric deflection and flow, and tilted thrust vector. For this analysis configuration, the abilities to handle an arbitrary joint clearance and an asymmetric flow by the present coupled framework will be verified.

Beyond that, realistic nozzle configuration was introduced. The one-way and both-way coupled analyses were performed on realistic configuration (Figure 13). In these analyses, all revolute joints were restrained to maintain the nozzle shape. In practice, the nozzle shape is controlled by actuators. DYMORE provides actuator and controller element to simulate closed-loop control. As a preliminary analysis, only a single torsional spring was adopted and other joints were released to predict realistic behavior (Figure 14). These analyses are summarized in Table 2.
Table 2. Computational configurations for coupled analysis.

<table>
<thead>
<tr>
<th>Configuration</th>
<th>Configuration1</th>
<th>Configuration2</th>
<th>Configuration3</th>
<th>Configuration4</th>
<th>Configuration5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coupled analysis type</td>
<td>One-way coupling</td>
<td>Both-way coupling</td>
<td>One-way coupling</td>
<td>Both-way coupled analysis</td>
<td></td>
</tr>
<tr>
<td>Joints type</td>
<td>N/A</td>
<td>Joint with clearance</td>
<td>Fixed joints</td>
<td>Torsional spring and released joints</td>
<td></td>
</tr>
</tbody>
</table>

Figure 15. Deformed geometry of the nozzle inner wall.

5.2 Simple nozzle one-way coupled analysis (Configuration 1)

The steady-state flow analysis was performed for the simple nozzle configuration. Such CFD pressure result was transformed to the dimensional distributed loads on the nozzle wall. Then nonlinear static structural analysis was performed by NASTRAN, and the resulting strain predictions were compared with those obtained by the present analysis. As illustrated in Figures 15, good agreement upon the deflection was observed. The maximum difference in the deflection was 4.49 $\times$ $10^{-3}$ m at the end of the nozzle, and the average difference was 8.5%.

5.3 Simple nozzle with joint clearance (Configuration 2)

Both-way coupled analysis was performed for the two-dimensional simple nozzle including an imaginary joint clearance. The symmetric CFD analysis was performed first. The symmetric nozzle wall pressure was delivered to DYMORE which included a clearance element. After nonlinear structural analysis was performed in DYMORE, the asymmetric deflection result was delivered to back CFD analysis. Such intrinsic result exchanges were conducted for several times.

The computation and result exchange between CFD and DYMORE were conducted seven times for the present asymmetric flow. Each step was 0.01 second long, and thus the total time for the present analysis time would be 0.07 second. The pressure difference at the seventh step between the upper and lower wall was 11.2%. Also, kink of 1.69% relative to nozzle inner radius was observed at the center of the clearance joint in the upper half component. As a result, thrust vector was found to be tilted by 1.77° because of asymmetric deflection and the flow as illustrated in Figure 16. A straight solid line which designates the thrust vector was obtained along the locations where higher Mach number was observed than those in the surroundings.

5.4 Realistic nozzle with restrained joints (Configurations 3, 4)

The one-way and both-way coupled analyses were performed with the realistic two-dimensional nozzle configuration. The realistic nozzle configuration consists of multiple beams and revolute joints without any clearance. For the present configuration, rotation on the joints was restrained to maintain the nozzle shape. The present flow analysis was capable of predicting the
temperature on the inner and outer wall. Heat conduction analysis was conducted by NASTRAN to obtain the temperature results on the entire components.

The interface module produced DYMORE input, which included distributed load and material properties. Nonlinear flexible multi-body dynamic analysis was conducted. For both-way coupled analysis, structural deflection induced at the nozzle wall was deliver to flow analysis. Total strain result was obtained as a summation of the mechanical and thermal strain. Figure 17 illustrates the mechanical and thermal strain results in the realistic nozzle configuration. Restriction of relative rotation with respect to the joint induced similar mechanical strain result of both-way coupled analysis as those obtained by the one-way coupled analysis.

5.5 Realistic nozzle with a single torsional spring (Configuration 5)

Restriction of relative rotation with respect to the joint induced similar strain results in one-way and both-way coupled analyses. For realistic configuration, single torsional spring was added instead of joint and both-way coupled analysis was conducted. Figure 18 shows the mechanical strain result in the realistic nozzle configuration with a single torsional spring. The maximum mechanical strain was reduced by 41% when compared with those obtained by both-way analysis for the realistic nozzle with restrained joints. Such reduction in the strain is due to the fact that the revolute joints in the configuration were now released to rotate freely.

6 CONCLUSIONS

In the present paper, one-way and both-way coupled analyses for thermo-fluid-flexible multi-body dynamics were performed for two-dimensional engine nozzles. DYMORE was employed for structural analysis, and flow solver with mixed unstructured grid was developed. Heat conduction and thermal analysis was conducted by NASTRAN. The interface module among these analyses was devised for result exchanges. Then, the thermo-fluid-flexible multi-body dynamic coupled analysis framework for a jet engine nozzle was established.

To validate the present combined analytical procedure, simplified two-dimensional nozzle configuration was devised. One-way coupled analysis was performed to combine CFD analysis and DYMORE. The strain result showed good agreement, within less than 10% difference compared to those by NASTRAN. Thus, the present coupled analysis procedure was validated.

In order to simulate the asymmetric flow pattern introduced by the joint clearance, both-way coupled analysis was performed. Imaginary clearance was added to the joints in the upper half component in the two-dimensional simplified nozzle. Iterative analysis was performed seven times for such asymmetric flow. As a result, 11.2% average pressure difference between the upper and lower-half components and kink of 1.69% relative to nozzle inner radius were obtained. And the thrust vector was tilted by 1.77°.
The total strain results for the realistic nozzle configuration were predicted with one-way and both-way coupled analyses. Thermal analyses were conducted by NASTRAN. The total strain was obtained as a summation of the mechanical strain and thermal strain. However, all revolute joints were restrained to maintain the nozzle shape, and similar strain results between one-way and both-way coupled analyses were induced. Finally, single torsional spring was introduced instead of fixed joint and both-way coupled analysis was conducted.

In future, prescribed actuation will be applied to the nozzle configuration. The heat convection and radiation effect of flow analysis will be included. Also, the present framework will be applicable for a three-dimensional nozzle configuration. The present beam elements and distributed loads will be replaced by the shell elements and pressure distributions. The element library of DYMORE already includes various actuator, shell element, and distributed pressure input function. For a three-dimensional nozzle analysis, parallel computation will be required in order to handle increased-sized fluid-structural interaction problem matrices. FETI (finite element tearing and interfacing) method will be employed for parallel computation.
element tearing and interconnecting) method is already implemented in a parallel version of DYMORE. Thus, it will be possible to build an analytical procedure for a three-dimensional nozzle configuration by extending from the present framework.

ACKNOWLEDGEMENT

This work was supported by Agency for Defense Development in Republic of Korea under the contract UD120039CD. Also, this work was supported by Advanced Research Center Program (NRF-2013R1A5A1073861) through the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIP) contracted through Advanced Space Propulsion Research Center at Seoul National University.

REFERENCES

Dynamic modeling of a 3D printer based on a four arms SCARA mechanism

Eduardo Paiva Okabe*, Pierangelo Masarati#

* School of Applied Sciences
University of Campinas - UNICAMP
Rua Pedro Zaccaria, 1300, PO Box 1068,
Limeira, Brazil
eduardo.okabe@fca.unicamp.br

# Dipartimento di Scienze e Tecnologie Aerospaziali
Politecnico di Milano
Via La Masa 34, 20156, Milano, Italy
pierangelo.masarati@polimi.it

Abstract
This work presents the dynamic simulation of a four arms SCARA (Selective Compliance Articulated Robot for Assembly) mechanism used in 3D printers in a multidisciplinary free software. Different extruder heads and motor supply voltage were simulated to show their impact on the construction of the printed part. To do the complete analysis of the printer, it is necessary to simulate the workflow to print a part. The steps of this workflow are part modeling, G-code generation, G-code translation, inverse kinematics analysis, motion translation, and dynamic analysis. After accomplishing these steps, the computation of the positioning error completes the analysis. The simulation showed that the supply voltage has the greater influence on the construction of the part. The extruder mass becomes relevant when the voltage is reduced. Simulation of the complete system also showed that electrical and mechanical components can be integrated in one model, although the behavior of components of one domain can restrict the simulation performance of the entire system.

1 INTRODUCTION
Nowadays, companies have been facing shorter and shorter time-to-market cycles. Their success is strictly related to their ability to conceive new products and services. In this environment, the application of new tools to accelerate the innovation process is fundamental to rapidly develop new products. The computational tools for product development linked to the manufacturing design helped to forge the “digital fabrication” concept. The basis of this concept is the application of software for three dimensional modeling (CAD), computational simulation (CAE), code generation (g-code) for CNC machines (CAM - computer aided manufacturing) and rapid prototyping process.

One of the most popular methods of rapid prototyping is the fused deposition method (FDM). It was developed at the end of the 1980s; its basic operation is to add melted plastic directly to the model through an extrusion head. The prototype is built by layers; therefore, the extruder head has to perform a series of repetitive motions until the layer is finished, before moving to the next one. The deposition process has to be relatively fast to assure the adhesion of the new layers on top of the former ones. The control of the motion of the deposition head along with the extrusion speed is fundamental to make the process precise and accurate.

The most common mechanism employed to move the deposition head is the gantry (Cartesian) mechanism. However, other types of mechanism such as Delta (parallel) and SCARA (parallel, redundant, selectively compliant) [5, 6] have been adopted in open source 3D printers because their fabrication and assembly is usually simpler and faster than Cartesian mechanisms. In fact, the latter usually require a more complex support frame. The reduced inertia of the mechanism is another advantage over gantry mechanisms, which makes increased deposition head speed possible and allows the construction of thin wall parts.

The four arms design (five bar mechanism) as shown in Figure 1 has been chosen to be selectively more rigid and stable compared to the traditional two arms concept. Another interesting feature
of the four arms design is that both motors can be attached to the support structure, whereas in the
traditional two arms configuration one of the motors must be attached to the elbow of the robotic arm.

The SCARA concept (Selective Compliance Articulated Robot for Assembly) was first introduced in Japan in 1979 [2]. This robot arm was designed to move fast in the horizontal plane with some compliance, essentially lumped in the joints, and with high stiffness regarding vertical motion. It has a small footprint compared to Cartesian robots, which renders it very useful for operations in restricted spaces.

In a 3D printer, the arms are driven by stepper motors, which decreases the need of feedback control, particularly in low speed operation. This represents a cheaper solution compared to servomotors, which are more commonly used in industrial robots.

To simulate the behavior of the SCARA mechanism associated with a set of stepper motors, a model was assembled using the free multibody software MBDyn [1]. The motor model proposed by Morar [3] was implemented in a user-defined module and further integrated in the complete mechanism model.

MBDyn features an inverse kinematic and dynamic analysis which is particularly interesting for the study of this kind of mechanism, since it translates the desired trajectory of the deposition head into joint rotations, including the case of redundant mechanisms, providing an estimate of the required torques.

Figure 2. Workflow of the dynamic analysis of the mechanism.

Figure 2 shows the workflow of the simulation performed to analyze the mechanism.

The first step is the modeling of the part that is going to be fabricated in a solid modeler. Then, a slicing software, such as Slic3r [7], is employed to generate the G-code program. This program tells the printer how it should move to build the part layer by layer. This slicing software takes into consideration some machine limits such as maximum printable dimensions, number of extrusion heads, maximum speed and resolution.

Normally, the firmware of the printer has to translate the desired motions of the deposition head into a sequence of motor steps. In this work, a Python script interprets the generated G-code, and provides the correct displacement, velocity and acceleration of the deposition head to MBDyn’s inverse kinematics analysis.

At the end of the inverse kinematics analysis, a file is generated with the rotation of each arm, which can be directly translated into the motion of the stepper motors. Another script interprets the
continuous movement provided by the former analysis into discrete steps that need to be performed by each of the motors. This phase is called motion discretization; it yields a file with time and direction of each motor step.

The dynamic simulation of the model considers rigid arms, but the stepper motors are modeled to show their influence on the dynamic behavior of the deposition mechanism. Even though the discrete motions are relatively close to the actual trajectory to be followed by the deposition head, the inertia of the components and the stepper motor behavior deviates the motion of the deposition head with respect to the desired path.

2 METHODOLOGY

The SCARA based printer is a mechanical/electrical system that can be formulated as a system of Differential-Algebraic Equations (DAE) [1]:

\[
\dot{\beta} + \left( \frac{\partial \phi}{\partial q} \right)^T \lambda_\phi + \left( \frac{\partial \psi}{\partial \dot{q}} \right)^T \lambda_\psi = \sum f(q, \dot{q}, t), \tag{1}
\]

where \( M \) is the inertia matrix, \( \beta \) is the vector of momentum and momenta moments, \( q \) is the generalized position vector, \( \phi \) is the system of holonomic constraint equations, \( \psi \) is the system of non-holonomic constraint equations, vectors \( \lambda_\phi \) and \( \lambda_\psi \) collect the Lagrange multipliers respectively associated with the holonomic and non-holonomic constraints, \( f \) is the vector of external loads, and \( (\dot{\cdot}) \) represents the time derivative of \( (\cdot) \).

Dynamic systems in MBDyn are based on nodes, in analogy with displacement-based Finite Element algorithms. Nodes provide degrees of freedom; they can be associated with different physical domains, such as mechanical, hydraulic, thermal and electrical. This structure makes it easier to integrate heterogeneous components of the same machine in one simulation.

For example, the stepping motor model presented in this work integrates dynamic structural and electric nodes in one element, which simulates the interaction between the mechanical and electrical domains.

Constraints, forces, voltages and currents have to be applied on, or imposed to, nodes; Thus, they become part of the modeled system. This is accomplished by adding two sets of equations to variables associated with each node. The first set is used in the assembly of the system of equations (1), and the second set is used during the nonlinear solution phase, if it is necessary, and it requires the analytical determination of the Jacobian matrix of Eq. (1).

2.1 Stepper motor formulation

The formulation of the stepper motor requires a combination of electric and structural nodes. The electric nodes receive the voltage coming from the stepper driver, which is controlled by a input signal that provides the step information (time and direction).

A stepper motor can have multiple phases; each phase has to be associated with an electric node, which has the following equation [3]:

\[
\epsilon_k = V_k - R \cdot i_k - L \frac{di_k}{dt} - e_k = 0 \tag{2}
\]

where \( V_k \) is the voltage applied by the stepper driver to the phase \( k \), \( R \) is the resistance of the winding connected to this phase, \( i_k \) is the current in the circuit of this phase, \( L \) is the winding
inductance of phase $k$, $d i_k / dt$ is the time derivative of the current, and $e_k$ is electromotive force induced.

The electromotive force $e_k$ varies with angular velocity of the stepper rotor:

$$e_k = K_m \sin (N_r \theta_z + \theta_{0k}) \cdot \omega_z$$  \hfill (3)

where $K_m$ is the motor constant, $N_r$ is the number of pole pairs, $\theta_z$ is the rotor rotation, $\theta_{0k}$ is the angular position of the $k$th winding in the stator, and $\omega_z$ is rotation speed of the rotor.

The electromagnetic torque is applied to the structural nodes; its magnitude is calculated according to the formula:

$$T_e = \sum_{k=1}^{s} K_m \sin (N_r \theta_z + \theta_{0k}) \cdot i_k$$  \hfill (4)

where $T_e$ is the electromagnetic torque, $n$ is the number of phases, and $i_k$ is the phase current.

The torque generated by the motor is applied to the rotor and, therefore, there is a corresponding reaction moment in the motor stator. The structural nodes attached to the rotor and the stator are defined in absolute coordinates, thus the torque has to be transformed from local to global coordinates:

$$C_1 = - R_1 k T_e$$  

$$C_2 = - C_1$$  

$$R_{1r} = R_1 \tilde{R}_1$$  \hfill (5)

where $C_1$ is the vector of the moment applied to the stator node, $k = \{001\}^T$ is a unit vector pointing along direction $Z$ (third coordinate), $C_2$ is the moment applied to the rotor node, $R_1$ is the orientation matrix associated with the stator node, and $\tilde{R}_1$ is the relative orientation matrix of the motor in relation with the stator node.

The Jacobian matrix is used by the implicit integration method employed by MBDyn. Therefore, its calculation is necessary for the analysis of the model. A perturbation can be applied to Eq. (2) to determine the component to be included into the Jacobian matrix of the complete system:

$$\delta \epsilon_k = \delta V_k - R \delta i_k - \delta \left( \frac{d i_k}{d t} \right) - \delta e_k = 0$$  \hfill (6)

The voltage $V_k$ is defined by the stepper driver and, therefore, its value is known and its virtual value $\delta V_k$ is equal to zero. The perturbation of the electromotive force is:

$$\delta e_k = K_m (N_r \cos (N_r \theta_z + \theta_{0k}) \omega_z \delta \theta_z + \sin (N_r \theta_z + \theta_{0k}) \delta \omega_z)$$  \hfill (7)

where the perturbation of the rotation speed $\omega_z$ is:

$$\delta \omega_z = k^T \delta R^T_{1r} [(\omega_2 - \omega_1) \times] \delta g_1 + k^T R^T_{1r} (\delta \omega_2 - \delta \omega_1)$$

$$= h b_0 k^T R^T_{1r} (\omega_2 - \omega_1) \times [1 - h b_0 (\omega_2 \times)] \delta g_1$$  \hfill (8)

The rotation angle $\theta_z$ is determined by the orientation matrices of the rotor and stator nodes:
\[ \theta = \alpha x \left( \exp^{-1} \left( R_{1r}^T R_2 \right) \right) \]
\[ \theta_c = k \cdot \theta \]

where \( \theta \) is the vector of the Euler parameters that represents the relative orientation between rotor and stator, \( (\cdot)^T \) represents the transpose of \( (\cdot) \), \( \alpha x \) is the inverse of the operator that generates the cross product matrix, \( \exp^{-1} \) is the operator that determines the Euler parameters from an orientation matrix, and \( R_2 \) is the orientation (rotation) matrix associated with the rotor node.

The perturbation of the rotation angle \( \delta \theta_c \) can be obtained through the orientation matrices:

\[ \delta \theta_c = k^T \Gamma(\theta)^{-1} R_{1r}^T (\theta_{2\delta} - \theta_{1\delta}) \]
\[ = h b_0 k^T \Gamma(\theta)^{-1} R_{1r}^T (\delta \dot{g}_2 - \delta \dot{g}_1) \]  

where \( \Gamma(\cdot) \) is the differential operator associated with the exponential map, \( \theta_{2\delta} \) is the virtual rotation vector of the rotor node, and \( \theta_{1\delta} \) is the virtual rotation vector of the stator node.

The perturbation of the moment applied to the rotor node can be used to determine the perturbation of the stator node, thus:

\[ \delta C_1 = \delta R_{1r} [3] T_e + R_{1r} [3] \delta T_e \]
\[ R_{1r} [3] = R_{1r} k \]  
\[ \delta T_e = K_m \sum_{k=1}^{n} \left( \sin(N_r \theta_2 + \theta_{3k}) \delta i_k + N_r \cos(N_r \theta_2 + \theta_{3k}) i_k \delta \theta_c \right) \]

Applying the updated-updated method [1], i.e. incrementally expressing rotations from their predicted value, such that the unknown rotation only accounts for the correction contribution, the virtual rotation term can approximated by:

\[ \delta R_{1r} [3] T_e = -R_{1r} [3] T_e \times \tilde{\delta g}_1 \]  

where \( \tilde{\delta g}_1 \) is the vector of virtual Cayley-Gibbs-Rodriguez parameters associated with the rotation of the stator node.

Using Eq. (12), and replacing the virtual perturbations of the parameters with their virtual velocities, Eq. (11) can be rewritten as:

\[ \delta C_1 = -h b_0 [R_{1r} [3] T_e \times] \tilde{\delta g}_1 + R_{1r} [3] h b_0 K_m \sum_{k=1}^{n} \left( \sin(N_r \theta_2 + \theta_{3k}) \delta \frac{d i_k}{d t} + N_r \cos(N_r \theta_2 + \theta_{3k}) i_k k^T \Gamma(\theta)^{-1} R_{1r}^T (\delta \dot{g}_2 - \delta \dot{g}_1) \right) \]
\[ = -h b_0 [R_{1r} [3] T_e \times] \tilde{\delta g}_1 + R_{1r} [3] h b_0 K_m \sum_{k=1}^{n} \left( \sin(N_r \theta_2 + \theta_{3k}) \delta \frac{d i_k}{d t} \right) + h b_0 K_m \sum_{k=1}^{n} \left( N_r \cos(N_r \theta_2 + \theta_{3k}) i_k \right) \cdot R_{1r} [3] \otimes \left( k^T \Gamma(\theta)^{-1} R_{1r}^T \right) (\delta \dot{g}_2 - \delta \dot{g}_1) \]  

2.2 Integration of elements related to different domains

The mathematical model of each part has to be integrated into the system; then, the complete simulation can be run. The main entities of integration are the elements, because they provide the
equations that express the interaction between domains. Figure 3 shows the system of one stepper motor, where the discrete motion constitutes the system input that is calculated from the inverse kinematic analysis. The stepper driver transforms the input signal into tension on the electric nodes that represent the motor phases.

The stepper motor is connected to electric and structural nodes; therefore, it relates the electrical and mechanical subsystems. This is a two way connection, which is shown by Eqs. (2) and (4), where voltage and current of the electric nodes generate torque between rotor and stator, and the motion of the rotor influences the current that is consumed by the motor.

The “body” elements are connected to the structural nodes of the stepper; they attribute inertia properties to the rotor and the stator. These nodes are constrained by a “total joint” [9], that allows the relative rotation of the rotor inside the stator. This joint completes the modeling of the stepper motor in MBDyn.

3 RESULTS

Figure 4 shows the fused deposition printer based on the SCARA geometry and a diagram of the machine developed in this work. The motion of the stepper motors is transmitted to concentric shafts by a set of pulleys. The upper end of the shafts is connected to the arms of the SCARA mechanism in the origin point of the coordinate system. The ends of both arms are connected to the extrusion head, thus the combined motion of each motor makes the head move in the xy plane. The motion of the deposition bed (table) is performed by other stepper motor, which only moves when the deposition of a new layer of the model needs to start.

The use of concentric shafts is a way to avoid locking positions during the printing process. The locking position occurs when the arm segments connected to the head form an angle of $180^\circ$.

Figure 3. Stepper motor system

Figure 4. RepRAP Morgan [5] (left) and SCARA model simulated (right).
Table 1. Scara printer parameters

<table>
<thead>
<tr>
<th>Part</th>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>arm (one segment)</td>
<td>mass</td>
<td>0.184 kg</td>
</tr>
<tr>
<td></td>
<td>moment of inertia</td>
<td>$I_{xx} = 3.07 \times 10^{-5}$ kg/m²</td>
</tr>
<tr>
<td></td>
<td>moment of inertia</td>
<td>$I_{yy} = 6.2 \times 10^{-4}$ kg/m²</td>
</tr>
<tr>
<td></td>
<td>moment of inertia</td>
<td>$I_{zz} = 6.4 \times 10^{-4}$ kg/m²</td>
</tr>
<tr>
<td></td>
<td>length</td>
<td>200 mm</td>
</tr>
<tr>
<td>deposition head</td>
<td>mass</td>
<td>0.12 kg</td>
</tr>
<tr>
<td>(lighter version)</td>
<td>moment of inertia</td>
<td>$I_{xx} = 4.52 \times 10^{-5}$ kg/m²</td>
</tr>
<tr>
<td></td>
<td>moment of inertia</td>
<td>$I_{yy} = 4.52 \times 10^{-5}$ kg/m²</td>
</tr>
<tr>
<td></td>
<td>moment of inertia</td>
<td>$I_{zz} = 1.35 \times 10^{-5}$ kg/m²</td>
</tr>
<tr>
<td>deposition head</td>
<td>mass</td>
<td>0.5 kg</td>
</tr>
<tr>
<td>(heavier version)</td>
<td>moment of inertia</td>
<td>$I_{xx} = 6.21 \times 10^{-4}$ kg/m²</td>
</tr>
<tr>
<td></td>
<td>moment of inertia</td>
<td>$I_{yy} = 2.05 \times 10^{-3}$ kg/m²</td>
</tr>
<tr>
<td></td>
<td>moment of inertia</td>
<td>$I_{zz} = 5.21 \times 10^{-4}$ kg/m²</td>
</tr>
<tr>
<td>stepper motor</td>
<td>rotor mass</td>
<td>0.07 kg</td>
</tr>
<tr>
<td></td>
<td>rotor moment of inertia</td>
<td>$5.4 \times 10^{-6}$ kg/m²</td>
</tr>
<tr>
<td></td>
<td>number of phases</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>winding inductance</td>
<td>10 mH</td>
</tr>
<tr>
<td></td>
<td>winding resistance</td>
<td>7.4 Ω</td>
</tr>
<tr>
<td></td>
<td>number of teeth $N_r$</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td>motor constant $K_m$</td>
<td>0.04 · $N_r$</td>
</tr>
<tr>
<td></td>
<td>viscous friction</td>
<td>0.5 kg·rad/s</td>
</tr>
<tr>
<td></td>
<td>step size</td>
<td>1.8°</td>
</tr>
<tr>
<td>reduction (motor to arm)</td>
<td>gear ratio</td>
<td>1:7.5</td>
</tr>
</tbody>
</table>

Table 1 shows some parameter used in the simulation. The motor model is based on a high torque two phase hybrid stepping motor (NEMA23), that has 200 steps per turn (=1.8° step) and a holding torque of 90 N·cm. The pulley sets that connect each stepper motor to its respective arm have a gear ratio of 1:7.5, which reduces the maximum speed, but improves the system precision. To make the motion even finer, the stepper driver adopts a microstepping strategy with a 1/16th step, thus the motor original step is divided by 16, which yields a step of 0.1125°, and a final arm rotation of 0.015° ($2.618 \times 10^{-4}$ rad).

All segments of both arms have the same length (200 mm), which is a little longer than RepRAP Morgan (150 to 170 mm). Each segment was modeled as a rigid body, that is connected to other segments by “total joint” elements that mimic revolute joints. The element “gear joint” [4] connects the motors with the arms. This element is able to reproduce the mechanical gearing reduction of the set of pulleys.

Also the extruder head was modeled as a rigid body; two versions were simulated. The “lighter” version is based on a hot end composed by a heater and a radiator. Due to its low inertia, this model is usually adopted in smaller 3D printers, that have less powerful motors, so they can move the head with less effort. The “heavier” version is a complete extruding system [11] that includes the feeding mechanism, thus it can pull the filament directly to the hot end, which allows a much better control of the extrusion flow.

To simulate the operation of the 3D printer, the manufacturing process of a simple cylinder with a closed bottom has been chosen. The STL model of the cylinder was converted by the software Slic3r [7] into G-code, which is compatible with all 3D printers based on the RepRap project. The G-code was translated into the extrusion head movement by a Python script based on the Marlin firmware [8]. This algorithm limits the speed in the intersections using a trapezoid strategy, i.e.,
the machine accelerates the extruder until it reaches the nominal speed determined in the G-code file; when it approaches an intersection, where the motion direction usually changes, the machine needs to decelerate to the established intersection speed. This strategy reduces the error caused by sudden changes of direction, where inertia continues to push the system into the original direction.

This script generates a trajectory for the extruder that is used by the inverse kinematic analysis of MBDyn to convert the extruder motion into joints rotation. The resulting joint rotation is processed by a second Python script that transforms the rotation in steps and their directions, corresponding to the movement that needs to be accomplished by each motor. This script must consider the reduction between arm and motor, and the strategy defined in the stepper driver. It also considers what motor position is closest to the position required by the trajectory. The generated files are the inputs to each of stepper motor systems used in the dynamic simulation of the SCARA printer.

The simulation of the printer produces the data regarding the dynamic behavior of all components. The final objective of this work is to verify the impact of the printer parameters on the part construction. To this end, the simulated motion of the extruding head must be compared with the planned trajectory, with the help of an error function. This function can be calculated using the approach adopted in [10], where the positioning error \( \varepsilon_{\text{pos}} \) is the distance between the planned and the actual position of the extruding head:

\[
\varepsilon_{\text{pos}} = \sqrt{(x_{\text{actual}} - x_{\text{planned}})^2 + (y_{\text{actual}} - y_{\text{planned}})^2}
\] (14)

The coordinate \( z \) is not considered in this calculation, because it represents the movement of the deposition bed, which is slow and is only performed upon completion of each layer.

The distribution of the error can be determined by clustering the errors between the planned and the actual position for every time instant calculated in the simulation. The number of occurrences (frequency) is normalized by the total number of steps of the simulation.

The process of manufacturing of the chosen part takes 274.8 seconds, which includes the initial positioning of the head and a previous deposition, that is usually employed to verify whether the extruder is working correctly. MBDyn took about 12 minutes to run the simulation on an Intel Core i7 (2670) computer. A variable time step strategy was adopted; most of the time the step was \( 10^{-3} \) seconds. The simulation required an average of 2.8 Jacobian evaluations per step.

### 3.1 Extruder head

In the first two simulations, the voltage supplied to the motors was set to 12 V. This voltage enables the motors to work with a higher torque; therefore, errors should be smaller when compared to lower supply voltages.

Figure 5 shows the error distribution curve and the trajectory deviation of the printer using the lighter extruding head and a stepper supply voltage of 12 V. The error distribution shows that most of positioning errors falls between 0 and 100 \( \mu \text{m} \) with a average of 46.1 \( \mu \text{m} \). The largest deviation in trajectory occurs in three different areas; two of them have been highlighted in Figure 5 (right).

The deviation in these areas is greater (almost 1.0 mm), because the head is accelerating or decelerating from high speed displacements, which occurs when the deposition is finished in a particular area and the head has to be quickly repositioned to a new deposition area.

Figure 6 (left) shows that for the heavier head the error distribution is very similar to the distribution presented in Figure 5, which means that the motor torque is enough to hold the increase of head inertia. There is a slight change in the average error (47.4 \( \mu \text{m} \)), but considering that the head mass increased four times, it is worth changing the extruder with a heavier version and gain more control over the deposition flow. Figure 6 (right) confirms that increasing the mass had a small influence on the positioning error.

Figure 7 compares the performance of both extruders in the areas denoted by Figures 5 and 6.
Figure 5. Error distribution (left) and trajectory deviation (right) of the machine with the lighter head (0.12 kg) and a stepper supply voltage of 12V.

Figure 6. Error distribution (left) and trajectory deviation (right) of the machine with the heavier head (0.5 kg) and a stepper supply voltage of 12V.

Figure 7. Comparison between the lighter and heavier extruders (left: Area 1; right: Area 2).
Although the trajectories are not identical, the deviation amplitudes are quite similar. The heavier model seems to take longer to reach the position determined by the stepper motor. Another problem is that the motion of the stepper motor is discrete (by steps), thus the trajectory is not always coincident with the motor rotation. This problem is mitigated by the transmission reduction and microstepping, but not eliminated.

### 3.2 Stepper voltage supply

The results show that the voltage supply is enough to overcome an increase of the extruding head mass; therefore, it is interesting to verify what happens if the voltage supplied by the stepper driver is decreased. The two simulations presented in this section use 5V instead of 12V, with the same head models.

The distribution of the error shows a higher frequency in the range of 30 to 60µm in Figure 8 (left), when compared with that of the simulations using 12V. The average error is also higher (48.2µm), which reveals that less voltage translates into less torque to control the motion of the extruder.

Figure 9 (left) shows that the increase of the mass leads to larger errors, which is indicated by the increase of the frequency in the range from 100 to 220µm. The largest errors increased from 0.8 to 1.2 mm in the same transition points of the previous simulations. This can be explained by the
Figure 10. Comparison between the lighter and the heavier extruder (left: Area 1; right: Area 2).

Table 2. Trajectory deviation

<table>
<thead>
<tr>
<th>supply voltage [V]</th>
<th>head mass [kg]</th>
<th>extruding head deviation [$\mu$m]</th>
<th>mean</th>
<th>standard deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>0.12</td>
<td>48.2</td>
<td>26.5</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>0.5</td>
<td>58.7</td>
<td>42.2</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>0.12</td>
<td>46.1</td>
<td>22.7</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>0.5</td>
<td>47.4</td>
<td>24.5</td>
<td></td>
</tr>
</tbody>
</table>

The fact that inertia plays a greater influence on the head movement, and the stepper motors cannot provide the necessary torque to control the movement.

Figure 10 shows that the deviation amplitude of the heavier extruder is larger than that of the lighter one. The detail of Area 2 (Figure 10, left) shows that the error increases when the motion direction changes, which is visible when the head is moving to the next filling line.

3.3 Results summary

Table 2 shows the trajectory deviation for different voltages and head masses. The combination of low voltage and heavier extruder showed the worst performance of all simulations. The sample standard deviation of the error in this case is almost twice that of the best configuration (lighter head and higher voltage).

It can also be noticed that increasing the motor torque by using a higher voltage can be a good solution to compensate heavier extruders.

Table 3 shows the power supplied by the driver to the stepper motor. The power supplied when the voltage is set to 12V is five times higher than that with the 5V supply, which means that a more powerful driver is needed when the system operates at higher voltages. Open source printers usually adopt stepper drivers (e.g. A4988) that can deal with an average current of 1A with peaks of 2A, which is less than what is required by the lighter configuration (13 W $\rightarrow$ 2.6A).

In this case, the speed has to be limited, otherwise, the overcurrent protection system would cut the motor tension.

4 CONCLUSIONS

The modeling of the 3D printer based on a SCARA mechanism showed that it is possible to use a multidisciplinary platform to simulate such a coupled problem. Besides that, MBDyn was
also used to generate the control sequence of each stepping motor through the inverse kinematics analysis.

To simulate the printer motors, a mathematical model of a hybrid stepping motor was implemented in MBDyn, along with a stepper driver to control the motor motion. Although it is not easy to determine the analytical Jacobian matrix of these mathematical models, the advantage of having a smooth and relatively fast simulation justifies the effort required to implement it.

Electrical and mechanical components are usually hard to integrate, especially when the model incorporates digital components, such as the motor driver. Sudden changes of the input signal causes high frequencies that the differential solvers do not manage easily. The combination of the nonlinear implicit solver and a variable time step seemed to solve the problems of mixing components of different domains that operate at different rates.

The 3D printing process adds a level of difficulty to the simulation, because the modeling of the part to be built and its translation to machine movements require a previous knowledge of the equipment and material to be printed, and they change the way this translation is made.

The simulation of the dynamic behavior of the printer helps to understand what can be changed in the design of this machine. For instance, the effect of increasing the mass of the extruder head can be overcome by increasing the stepping motor supply. However, the driver must be capable of providing enough power to feed the stepper, otherwise the driver protection system cuts the current introducing a perturbation in the motion.

**ACKNOWLEDGEMENT**

This work was supported by CNPq (National Council for Scientific and Technological Development - Brazil).

**REFERENCES**


Index-2 Co-Simulation Approach for Solver Coupling with Algebraic Constraints

Daixing Lu*, Pu Li*, Bernhard Schweizer*

*Institute of Applied Dynamics
Technical University of Darmstadt
64287 Darmstadt, Germany
[lu, li, schweizer]@sds.tu-darmstadt.de

ABSTRACT

A co-simulation method on index-2 level is presented and analyzed with respect to numerical stability and convergence behavior. Therefore, two arbitrary mechanical subsystems are considered, which are coupled by algebraic constraint equations (i.e. by rigid joints). In this manuscript, we act on the assumption that each subsystem has its own solver, which integrates independently from the other subsystem solver. A weak coupling approach is used for coupling the two subsystem solvers. The coupling method analyzed here is based on the well-known stabilized index-2 formulation for multibody systems (GGL-formulation). The presented coupling technique is semi-implicit and based on a predictor/corrector approach.

Keywords: co-simulation, solver coupling, subcycling, algebraic constraints, predictor/corrector approach, index-2 formulation

1 STABILITY AND CONVERGENCE ANALYSIS

1.1 Co-Simulation Test-Model for Constraint Coupling

The definition of the numerical stability of time integration methods is based on Dahlquist´s test equation $\dot{y}(t) = \lambda \cdot y(t)$, where $\lambda = \lambda_r + i \lambda_i \in \mathbb{C}$ denotes an arbitrary complex parameter. From the mechanical point of view, Dahlquist´s test equation can be interpreted as the complex representation of the equations of motion of the homogenous linear mass-spring-damper oscillator. By discretizing the test equation with a linear time integration scheme, a linear recurrence equation is obtained. The stability of this recurrence equation defines the numerical stability of the underlying time integration scheme. Usually, 2D-stability plots are used for illustrating the stability behavior as a function of $h \lambda_r$ and $h \lambda_i$ ($h$ terms the integration step size).

In order to analyze the stability of co-simulation methods with constraint coupling, it is straightforward to use the linear two-mass oscillator depicted in Figure 1 as co-simulation test model. This system can be regarded as two Dahlquist equations – i.e. two linear single-mass oscillators (masses $m_1/m_2$, stiffness coefficients $c_1/c_2$, damping coefficients $d_1/d_2$) – coupled by the linear constraint equation $x_2 - x_1 = 0$.

Figure 1. Two-mass oscillator: Interpretation as two algebraically coupled Dahlquist equations.

With the position variables $x_1, x_2$, the related velocities $v_1, v_2$ and the Lagrange multiplier $\lambda_c$, the equations of motion of the co-simulation test model are given by the index-3 DAE system
Introducing the five parameters \( \bar{c}_1, \bar{d}_1, \alpha_{m21}, \alpha_{c21}, \alpha_{d21} \), Eq. (1) can be rewritten as

\[
\begin{align*}
\dot{x}_1 & = v_1 \\
\dot{v}_1 & = -\frac{c_1}{m_1} x_1 - \frac{d_1}{m_1} v_1 + \frac{\lambda_c}{m_1} \\
\dot{x}_2 & = v_2 \\
\dot{v}_2 & = -\frac{c_2}{m_2} x_2 - \frac{d_2}{m_2} v_2 - \frac{\lambda_c}{m_2} \\
0 & = x_2 - x_1.
\end{align*}
\] (1)

the dimensionless time \( \bar{t} = \frac{t}{h} \), the dimensionless velocities \( \bar{v}_1 = H \cdot v_1, \bar{v}_2 = H \cdot v_2 \) and the dimensionless Lagrange multiplier \( \bar{\lambda}_c = \frac{\lambda_c \cdot H^2}{m_1} \), Eq. (2) can be rewritten as

\[
\begin{align*}
\bar{x}_1' & = \bar{v}_1 \\
\bar{v}_1' & = -\bar{c}_1 \cdot \bar{x}_1 - \bar{d}_1 \cdot \bar{v}_1 + \bar{\lambda}_c \\
\bar{x}_2' & = \bar{v}_2 \\
\bar{v}_2' & = -\frac{\alpha_{c21}}{\alpha_{m21}} \cdot \bar{c}_1 \cdot \bar{x}_2 - \frac{\alpha_{d21}}{\alpha_{m21}} \cdot \bar{d}_1 \cdot \bar{v}_2 - \frac{1}{\alpha_{m21}} \cdot \bar{\lambda}_c \\
0 & = \bar{x}_2 - \bar{x}_1.
\end{align*}
\] (3)

In the above equation, \( x_1 \) and \( x_2 \) are assumed to be dimensionless position coordinates. Note that \( H \) terms the macro-step size of the co-simulation. \((\cdot)' = \frac{d(\cdot)}{d\bar{t}}\) specifies the derivative with respect to the dimensionless time \( \bar{t} \).

Discretizing Eq. (3) with a linear co-simulation scheme yields a homogenous linear recurrence equation system. The stability of the recurrence equation system can easily be determined by calculating the spectral radius \( \rho \) of the system. The recurrence system – and as a consequence the underlying co-simulation method – is stable, if \( \rho \leq 1 \). The spectral radius \( \rho \) is a function of the five independent test-model parameters defined in Eq. (2). In order to illustrate the stability behavior of co-simulation methods with 2D-stability plots, it is useful to fix 3 parameters so that \( \rho \) can be plotted as a function of the remaining 2 parameters.

1.2 Stability Analysis For Index-2 Co-Simulation Approach

Making use of the index-2 coupling approach of Ref. [18], the co-simulation test model is decomposed into two subsystems as illustrated in Figure 2. Two coupling variables are used to connect the subsystems: The Lagrange multiplier \( \lambda_c \), which characterizes the reaction force, and the additional Lagrange multiplier \( \mu_c \), which represents the projection term in the kinematical differential equations.

![Figure 2. Decomposed co-simulation test model: Coupling variables \( \lambda_c \) and \( \mu_c \).](image-url)
The equations of motion of the decomposed system read in dimensionless form

**Subsystem 1:**

\[ x'_1 = \ddot{v}_1 + \mu_c \]
\[ \ddot{v}_1' = -\varepsilon_1 \cdot x_1 - \ddot{d}_1 \cdot \ddot{v}_1 + \ddot{\lambda}_c \]  

**Subsystem 2:**

\[ x'_2 = \ddot{v}_2 - \mu_c \]
\[ \ddot{v}_2' = -\frac{a_{c21}}{a_{m21}} \cdot \ddot{v}_2 - \frac{1}{a_{m21}} \ddot{\lambda}_c \]  

**Coupling conditions:**

\[ g_{c\lambda} := x_2 - x_3 = 0 \]
\[ \bar{g}_{c\mu} := \ddot{v}_2 - \ddot{v}_1 = 0 \]  

Note that \( \bar{\mu}_c = H \cdot \mu_c \) denotes the dimensionless projection variable. In order to derive the recurrence equation system for the discretized test model, the general macro-step from \( \bar{T}_N \) to \( \bar{T}_{N+1} \) is considered \((\bar{T}_N = \frac{T_N}{H} \) terms the dimensionless macro-time point). For integrating the subsystems from \( \bar{T}_N \) to \( \bar{T}_{N+1} = \bar{T}_N + 1 \), the coupling variables \( \bar{\lambda}_c(\bar{t}) \) and \( \bar{\mu}_c(\bar{t}) \) have to be approximated in the time interval \([\bar{T}_N, \bar{T}_{N+1}]\). Here, constant approximation polynomials are used. Direct application of higher order approximation polynomials is not possible, since numerical instabilities are detected in this case.

At the beginning of the macro-step, the state variables and the coupling variables are assumed to be known

\[ x_1(\bar{t} = \bar{T}_N) = x_{1,N}, \quad \ddot{v}_1(\bar{t} = \bar{T}_N) = \ddot{v}_{1,N}, \]
\[ x_2(\bar{t} = \bar{T}_N) = x_{2,N}, \quad \ddot{v}_2(\bar{t} = \bar{T}_N) = \ddot{v}_{2,N}, \]
\[ \bar{\lambda}_c(\bar{t} = \bar{T}_N) = \bar{\lambda}_{c,N}, \quad \bar{\mu}_c(\bar{t} = \bar{T}_N) = \bar{\mu}_{c,N}. \]  

The index-2 coupling approach method of Ref. [18] is based on a predictor/corrector technique and is accomplished in three steps. In the following representation, predicted variables are indicated with an upper index \( p \) (e.g. \( x_{1,N+1}^p \)). Variables without upper index characterize corrected variables (e.g. \( x_{1,N+1} \)). \( \mathbf{z}_N = (x_{1,N}, \ddot{v}_{1,N}, x_{2,N}, \ddot{v}_{2,N})^T \) is an auxiliary vector, which contains the state variables of both subsystems at the macro-time points \( \bar{T}_N \).

**Step 1: Predictor Step**

- By analytically integrating subsystem 1 and subsystem 2 from \( \bar{T}_N \) to \( \bar{T}_{N+1} \) with the initial conditions (5a) and with the predictor (extrapolation) polynomials

\[ \bar{\lambda}_{c,N}^p(t) = \bar{\lambda}_{c,N} = \text{const.} \quad \text{and} \quad \bar{\mu}_{c,N}^p(t) = \bar{\mu}_{c,N} = \text{const.}, \]  

one gets the predicted state variables at the macro-time point \( \bar{T}_{N+1} \)

\[ x_{1,N+1}^p = x_{1,N+1}^p(\bar{\lambda}_{c,N}, \bar{\mu}_{c,N}, \mathbf{z}_N), \]
\[ \ddot{v}_{1,N+1}^p = \ddot{v}_{1,N+1}^p(\bar{\lambda}_{c,N}, \bar{\mu}_{c,N}, \mathbf{z}_N), \]
\[ x_{2,N+1}^p = x_{2,N+1}^p(\bar{\lambda}_{c,N}, \bar{\mu}_{c,N}, \mathbf{z}_N), \]
\[ \ddot{v}_{2,N+1}^p = \ddot{v}_{2,N+1}^p(\bar{\lambda}_{c,N}, \bar{\mu}_{c,N}, \mathbf{z}_N). \]  

**Step 2: Calculation of Corrected Coupling Variables**

- An analytical integration of subsystem 1 and subsystem 2 from \( \bar{T}_N \) to \( \bar{T}_{N+1} \) with the initial conditions (5a) and with the coupling variables
\[
\dot{\lambda}_c^*(t) = \lambda_{c,N+1}^* = \text{const.} \quad \text{and} \quad \dot{\mu}_c^*(t) = \mu_{c,N+1}^* = \text{const.}
\]  
\tag{8}

yields the following state variables at the macro-time point \( \bar{T}_{N+1} \)
\[
\begin{align*}
    x_{1,N+1}^* &= x_{1,N+1}^*(\lambda_{c,N+1}^*, \mu_{c,N+1}^*, z_N), \\
    v_{1,N+1}^* &= v_{1,N+1}^*(\lambda_{c,N+1}^*, \mu_{c,N+1}^*, z_N), \\
    x_{2,N+1}^* &= x_{2,N+1}^*(\lambda_{c,N+1}^*, \mu_{c,N+1}^*, z_N), \\
    v_{2,N+1}^* &= v_{2,N+1}^*(\lambda_{c,N+1}^*, \mu_{c,N+1}^*, z_N).
\end{align*}
\tag{9}
\]

Note that \( \lambda_{c,N+1}^* \) and \( \mu_{c,N+1}^* \) represent arbitrary coupling variables at the macro-time point \( \bar{T}_{N+1} \).

By differentiating the state variables of Eq. (9) with respect to \( \lambda_{c,N+1}^* \) and \( \mu_{c,N+1}^* \), we get the partial derivatives
\[
\begin{align*}
    \frac{\partial x_{1,N+1}^*}{\partial \lambda_{c,N+1}^*} &= \text{const.}, & \frac{\partial v_{1,N+1}^*}{\partial \lambda_{c,N+1}^*} &= \text{const.}, & \frac{\partial x_{2,N+1}^*}{\partial \lambda_{c,N+1}^*} &= \text{const.}, & \frac{\partial v_{2,N+1}^*}{\partial \lambda_{c,N+1}^*} &= \text{const.}, \\
    \frac{\partial x_{1,N+1}^*}{\partial \mu_{c,N+1}^*} &= \text{const.}, & \frac{\partial v_{1,N+1}^*}{\partial \mu_{c,N+1}^*} &= \text{const.}, & \frac{\partial x_{2,N+1}^*}{\partial \mu_{c,N+1}^*} &= \text{const.}, & \frac{\partial v_{2,N+1}^*}{\partial \mu_{c,N+1}^*} &= \text{const.}
\end{align*}
\tag{10}
\]

It should be mentioned that the partial derivatives are constant, since the state variables of Eq. (9) are only linear functions of \( \lambda_{c,N+1}^* \) and \( \mu_{c,N+1}^* \).

With the predicted variables and with the partial derivatives, it is possible to derive corrected coupling variables, which fulfill the coupling conditions (4c) at the macro-time point \( \bar{T}_{N+1} \). Considering the fixed time point \( \bar{T}_{N+1} \), \( g_{c\lambda,N+1} \) and \( g_{c\mu,N+1} \) can be regarded as functions of the coupling variables \( \lambda_{c,N+1}^* \) and \( \mu_{c,N+1}^* \), i.e.
\[
\begin{align*}
    g_{c\lambda,N+1}(\lambda_{c,N+1}^*, \mu_{c,N+1}^*) &= x_{2,N+1}^*(\lambda_{c,N+1}^*, \mu_{c,N+1}^*) - x_{1,N+1}^*(\lambda_{c,N+1}^*, \mu_{c,N+1}^*), \\
    g_{c\mu,N+1}(\lambda_{c,N+1}^*, \mu_{c,N+1}^*) &= \bar{v}_{2,N+1}^*(\lambda_{c,N+1}^*, \mu_{c,N+1}^*) - \bar{v}_{1,N+1}^*(\lambda_{c,N+1}^*, \mu_{c,N+1}^*).
\end{align*}
\tag{11}
\]

Due to the fact that the state variables \( x_{1,N+1}^*, x_{2,N+1}^*, \bar{v}_{1,N+1}^*, \bar{v}_{2,N+1}^* \) depend only linearly on \( \lambda_{c,N+1}^* \) and \( \mu_{c,N+1}^* \), Eq. (11) can be expressed as
\[
\begin{align*}
    g_{c\lambda,N+1}(\lambda_{c,N+1}^*, \mu_{c,N+1}^*) &= g_{c\lambda,N+1}(\bar{c}^p) + \left. \frac{\partial g_{c\lambda,N+1}}{\partial \lambda_{c,N+1}^*} \right|_{\bar{c}^p} \cdot (\lambda_{c,N+1}^* - \bar{c}^p_{c,N+1}) \\
    & \quad + \left. \frac{\partial g_{c\lambda,N+1}}{\partial \mu_{c,N+1}^*} \right|_{\bar{c}^p} \cdot (\mu_{c,N+1}^* - \bar{c}^p_{c,N+1}) \\
    &= (x_{2,N+1}^* - x_{1,N+1}^*) + \left( \frac{\partial x_{2,N+1}^*}{\partial \lambda_{c,N+1}^*} - \frac{\partial x_{1,N+1}^*}{\partial \lambda_{c,N+1}^*} \right)_{\bar{c}^p} \cdot (\lambda_{c,N+1}^* - \bar{c}^p_{c,N+1}) \\
    & \quad + \left( \frac{\partial x_{2,N+1}^*}{\partial \mu_{c,N+1}^*} - \frac{\partial x_{1,N+1}^*}{\partial \mu_{c,N+1}^*} \right)_{\bar{c}^p} \cdot (\mu_{c,N+1}^* - \bar{c}^p_{c,N+1}),
\end{align*}
\tag{12}
\]

\[
\begin{align*}
    g_{c\mu,N+1}(\lambda_{c,N+1}^*, \mu_{c,N+1}^*) &= g_{c\mu,N+1}(\bar{c}^p) + \left. \frac{\partial g_{c\mu,N+1}}{\partial \lambda_{c,N+1}^*} \right|_{\bar{c}^p} \cdot (\lambda_{c,N+1}^* - \bar{c}^p_{c,N+1}) \\
    & \quad + \left. \frac{\partial g_{c\mu,N+1}}{\partial \mu_{c,N+1}^*} \right|_{\bar{c}^p} \cdot (\mu_{c,N+1}^* - \bar{c}^p_{c,N+1}).
\end{align*}
\tag{13}
where the vector $\tilde{\varphi}^p = (\tilde{\lambda}^p_{c,N+1}, \tilde{\mu}^p_{c,N+1})$ contains the predicted coupling variables $\tilde{\lambda}^p_{c,N+1} = \tilde{\lambda}_{c,N}$ and $\tilde{\mu}^p_{c,N+1} = \tilde{\mu}_{c,N}$ at the macro-time point $\tilde{T}_{N+1}$.

- Setting $g_{\tilde{c}\lambda,N+1}(\tilde{\lambda}^*_c,N+1, \tilde{\mu}^*_c,N+1)$ and $\tilde{g}_{\tilde{c}\mu,N+1}(\tilde{\lambda}^*_c,N+1, \tilde{\mu}^*_c,N+1)$ equal to zero, we can compute the corrected coupling variables $\tilde{\lambda}^*_c,N+1$ and $\tilde{\mu}^*_c,N+1$, i.e.

$$
g_{\tilde{c}\lambda,N+1}(\tilde{\lambda}^*_c,N+1, \tilde{\mu}^*_c,N+1) = 0 \quad \Rightarrow \quad \tilde{\lambda}^*_c,N+1, \tilde{\mu}^*_c,N+1 \, .
$$

Note that for the reason of a clear representation, different variables have been used for the general coupling variables $\tilde{\lambda}^*_c,N+1, \tilde{\mu}^*_c,N+1$ at the time points $\tilde{T}_{N+1}$ and the corrected coupling forces $\tilde{\lambda}^*_c,N+1, \tilde{\mu}^*_c,N+1$, which represent the roots of Eq. (13).

**Step 3: Corrector Step**

- Making use of the corrected coupling variables $\tilde{\lambda}^*_c,N+1$ and $\tilde{\mu}^*_c,N+1$ from Eq. (13), an analytical integration of subsystem 1 and subsystem 2 from $\tilde{T}_{N}$ to $\tilde{T}_{N+1}$ with the initial conditions (5a) yields the corrected state variables

$$
x_{1,N+1} = x_{1,N+1}(\tilde{\lambda}^*_c,N+1, \tilde{\mu}^*_c,N+1, z_N), \\
v_{1,N+1} = v_{1,N+1}(\tilde{\lambda}^*_c,N+1, \tilde{\mu}^*_c,N+1, z_N), \\
x_{2,N+1} = x_{2,N+1}(\tilde{\lambda}^*_c,N+1, \tilde{\mu}^*_c,N+1, z_N), \\
v_{2,N+1} = v_{2,N+1}(\tilde{\lambda}^*_c,N+1, \tilde{\mu}^*_c,N+1, z_N) \, .
$$

The variables $\tilde{\lambda}^*_c,N+1$ and $\tilde{\mu}^*_c,N+1$ according to Eq. (13) are functions of the predicted position and velocity variables and also of the predicted multipliers. Replacing the predicted variables with the help of Eq. (6) and Eq. (7), Eq. (13) eventually results in a relationship of the form

$$
\tilde{\lambda}^*_c,N+1 = \tilde{\lambda}^*_c,N+1(\tilde{\lambda}_{c,N}, \tilde{\mu}_{c,N}, z_N) \, , \\
\tilde{\mu}^*_c,N+1 = \tilde{\mu}^*_c,N+1(\tilde{\lambda}_{c,N}, \tilde{\mu}_{c,N}, z_N) \, .
$$

Combining Eq. (14) and Eq. (15), we obtain the governing recurrence equation system

$$
x_{1,N+1} = x_{1,N+1}(\tilde{\lambda}^*_c,N+1, \tilde{\mu}^*_c,N+1, z_N), \\
v_{1,N+1} = v_{1,N+1}(\tilde{\lambda}^*_c,N+1, \tilde{\mu}^*_c,N+1, z_N), \\
x_{2,N+1} = x_{2,N+1}(\tilde{\lambda}^*_c,N+1, \tilde{\mu}^*_c,N+1, z_N), \\
v_{2,N+1} = v_{2,N+1}(\tilde{\lambda}^*_c,N+1, \tilde{\mu}^*_c,N+1, z_N),
$$

$$
\tilde{\lambda}^*_c,N+1 = \tilde{\lambda}^*_c,N+1(\tilde{\lambda}_{c,N}, \tilde{\mu}_{c,N}, z_N), \\
\tilde{\mu}^*_c,N+1 = \tilde{\mu}^*_c,N+1(\tilde{\lambda}_{c,N}, \tilde{\mu}_{c,N}, z_N) \, .
$$

Eq. (16) is a system of 6 coupled linear recurrence equations for the state variables and the Lagrange multipliers. With the auxiliary vectors $q_{N+1} = ...$
\[(x_{1,N+1}, \tilde{y}_{1,N+1}, x_{2,N+1}, \tilde{y}_{2,N+1}, \tilde{\lambda}_{C,N+1}, \tilde{\mu}_{C,N+1})^T \in \mathbb{R}^6, \quad q_N = \ldots, \] etc., which collect the state variables of both subsystems and the Lagrange multipliers at the macro-time points \(\tilde{T}_{N+1}, \tilde{T}_N,\) etc., Eq. (16) can symbolically be rewritten as

\[A_{N+1} \cdot q_{N+1} + A_N \cdot q_N = 0. \tag{17}\]

The matrices \(A_{N+1}, A_N \in \mathbb{R}^{6 \times 6}\) are real-valued and constant. They depend only on the 5 parameters of the co-simulation test model. The stability of the recurrence equation system (17) defines the numerical stability of the underlying co-simulation method. Stabilizing or destabilizing effects due to a numerical subsystem integration are not present, since the subsystems are solved analytically. The spectral radius \(\rho\), i.e. the magnitude of the largest eigenvalue, characterizes the stability of the recurrence equation system (17). The co-simulation approach is stable, if \(\rho \leq 1\).

For general nonlinear co-simulation problems, the 3-stage predictor/corrector approach represents a semi-implicit method. To realize a fully implicit co-simulation approach, a corrector iteration with several corrector steps has to be accomplished for nonlinear problems. For the linear test model, the partial derivatives, see Eq. (10), are constant so that the above approach represents a fully implicit method.

For illustrating the stability behavior of co-simulation methods, it is useful to replace the 5 independent test-model parameters of Eq. (2) by the following set of parameters

\[
\begin{align*}
\tilde{\lambda}_{r1} &= -\frac{d_1}{2}, \\
\tilde{\lambda}_{i1} &= \frac{1}{2} \sqrt{4 \cdot \tilde{c}_1 - d_1^2}, \\
\alpha_{m21} &= m_2 \frac{m_1}{m_2}, \\
\alpha_{d21} &= \frac{\alpha_d}{m_2}, \\
\alpha_{\tilde{c}21} &= \frac{\alpha_{\tilde{c}}}{m_2}, \\
\alpha_{\tilde{\lambda}_{i2}} &= \frac{1}{\alpha_{m21}} \sqrt{4 \cdot \alpha_{m21} \cdot \alpha_{\tilde{c}21} \cdot \tilde{c}_1 - \alpha_{d21}^2 \cdot d_1^2}.
\end{align*}
\tag{18}\]

The mechanical interpretation of these parameters is straightforward. \(\tilde{\lambda}_{r1}\) and \(\tilde{\lambda}_{i1}\) represent the dimensionless real and imaginary part of the eigenvalue of subsystem 1. The parameter \(\alpha_{m21}\) terms the mass ratio. \(\alpha_{\tilde{c}21}\) and \(\alpha_{\tilde{\lambda}_{i2}}\) denote the ratio of the real and the imaginary part of the dimensionless eigenvalue of subsystem 2 with respect to \(\tilde{\lambda}_{r1}\) and \(\tilde{\lambda}_{i1}\).

The spectral radius \(\rho\) is only a function of the five independent parameters of the co-simulation test-model. Fixing 3 parameters, \(\rho\) can be plotted in 2D-stability diagrams as a function of the remaining 2 parameters. In the following, the 3 parameters \(\alpha_{m21}, \alpha_{\tilde{c}21}\) and \(\alpha_{\tilde{\lambda}_{i2}}\) are fixed and 2D-stability plots are shown, where \(\rho\) is plotted as a function of \(\tilde{\lambda}_{r1}\) and \(\tilde{\lambda}_{i1}\) in the range \([-10,0]\) and \([0,10]\). The spectral radius \(\rho\) of the recurrence equation system (17) can only be computed numerically. The circles in the stability plots represent stable points, i.e. parameter configurations for which \(\rho \leq (1 + 10^{-10})\) holds. To minimize floating point errors, calculation of \(\rho\) has been accomplished with 128 decimal places so that points at the boundary of the stability region (\(\rho \approx 1\)) are computed very accurately.

In Figure 3, stability plots are collected for the seven parameter sets \((\alpha_{m21} = 1, \alpha_{\tilde{c}21} = 1, \alpha_{\tilde{\lambda}_{i2}} = 1), (\alpha_{m21} = 10, \alpha_{\tilde{c}21} = 1, \alpha_{\tilde{\lambda}_{i2}} = 1), (\alpha_{m21} = 1, \alpha_{\tilde{c}21} = 10, \alpha_{\tilde{\lambda}_{i2}} = 1), (\alpha_{m21} = 1, \alpha_{\tilde{c}21} = 1, \alpha_{\tilde{\lambda}_{i2}} = 10), (\alpha_{m21} = 5, \alpha_{\tilde{c}21} = 5, \alpha_{\tilde{\lambda}_{i2}} = 5), (\alpha_{m21} = 10, \alpha_{\tilde{c}21} = 10, \alpha_{\tilde{\lambda}_{i2}} = 10)\) and \((\alpha_{m21} = 100, \alpha_{\tilde{c}21} = 100, \alpha_{\tilde{\lambda}_{i2}} = 100)\). In the first four plots and in the sixth plot, only stable points are detected in the considered parameter range for \(\tilde{\lambda}_{r1}\) and \(\tilde{\lambda}_{i1}\). The fifth and seventh plot show some unstable points at the \(\tilde{\lambda}_{i1}\)-axis.
Figure 3. Stability plots for the index-2 co-simulation method for different test-model parameters.
1.3 Convergence Plots

Next, the local and global error of the index-2 co-simulation approach is analyzed. Therefore, convergence plots are calculated with the co-simulation test model. Four different numerical errors are distinguished. The relative global error \( \varepsilon_{\text{glo,pos}} \) for the position variables is calculated by the normalized root mean square error (NRMSE) according to

\[
\varepsilon_{\text{glo,pos}} = \sqrt{\frac{\sum_N (x_1(T_N) - x_{1,\text{mean}})^2 + \sum_N (x_2(T_N) - x_{2,\text{mean}})^2}{\sum_N (x_1(T_N) - x_{1,N})^2 + \sum_N (x_2(T_N) - x_{2,N})^2}} \quad (19)
\]

with 

\[
x_{1,\text{mean}} = \sum_N \frac{x_1(T_N)}{N_{\text{total}}}, \quad x_{2,\text{mean}} = \sum_N \frac{x_2(T_N)}{N_{\text{total}}}.
\]

The values \( x_{1,N}, x_{2,N} \) denote the co-simulation results at the macro-time points \( T_N \); \( x_1(T_N), x_2(T_N) \) represent the values of the analytical solution at the macro-time point \( T_N \). The total number of macro-steps is \( N_{\text{total}} \). The global error \( \varepsilon_{\text{glo,vel}} \) for the velocity variables \( v_1, v_2 \) and the related local errors \( \varepsilon_{\text{loc,pos}} \) and \( \varepsilon_{\text{loc,vel}} \) for the position and velocity variables are calculated in a similar manner. Note that the convergence plots have been computed with the test-model parameters \( m_1 = 1, m_2 = 2, c_1 = c_2 = 1000, d_1 = d_2 = 10 \).

Convergence plots illustrating the global and local errors \( \varepsilon_{\text{glo,pos}}, \varepsilon_{\text{glo,vel}}, \varepsilon_{\text{loc,pos}} \) and \( \varepsilon_{\text{loc,vel}} \) are shown in Figure 4. As can be observed, the global errors \( \varepsilon_{\text{glo,pos}} \) and \( \varepsilon_{\text{glo,vel}} \) converge with \( O(H^2) \), whereas the local errors \( \varepsilon_{\text{loc,pos}} \) and \( \varepsilon_{\text{loc,vel}} \) converge with \( O(H^3) \).

Figure 4. Convergence plot for the index-2 co-simulation approach: Local and global errors of the position and velocity variables over the macro-step size \( H \).

2 COUPLING ARBITRARY MULTIBODY SYSTEMS

Application of the index-2 co-simulation approach for coupling 2 arbitrary multibody subsystems by algebraic constraints is in detail described in Ref. [18]. As an example, we consider the classical planar four-bar linkage depicted in Figure 5. This system consists of three rigid links and four ideal revolute joints. To decompose the overall system into two subsystems, link 2 is split into two parts (\( m_{21} \) and \( m_{22} \)). The first subsystem contains link 1 (\( m_1 = 1 \) kg, \( J_1 = 1/12 \) kg \( m^2 \), \( l_1 = 1 \) m) and the left part of link 2 (\( m_{21} = 1 \) kg, \( J_{21} = 1/12 \) kg \( m^2 \), \( l_{21} = 1 \) m). The second subsystem is represented by the right part of link 2 (\( m_{22} = 1 \) kg, \( J_{22} = 1/12 \) kg \( m^2 \), \( l_{22} = 1 \) m) and by link 3 (\( m_3 = 2 \) kg, \( J_3 = 2/3 \) kg \( m^2 \), \( l_3 = 2 \) m). The two subsystems are coupled by a fixed joint as shown in the figure. The system is driven by the torque \( M_2(t) = 0.5 \cdot t \) Nm, which is applied at the center of mass of link 1. Gravity acts in
negative y-direction \((g = 9.81 \text{ m/s}^2)\). Subsystem 1 and 2 are both mathematically described by a DAE system, which has been integrated numerically with an implicit Runge-Kutta integrator using Baumgarte stabilization. As initial conditions, \(\varphi_{1,0} = \frac{\pi}{4} \text{ rad} \) and \(\dot{\varphi}_{1,0} = 0 \text{ rad/s} \) have been chosen. Simulation results are collected in Figure 6, where the displacements \(x_{21}(t)\) and \(y_{21}(t)\) of the center of mass \(S_{21}\) and the angle \(\varphi_{21}(t)\) are shown. Moreover, the coupling variables \(\lambda_{ca_x}(t), \lambda_{ca_y}(t)\) and \(\lambda_{cp}(t)\) acting at the fixed joint are shown. The figure also depicts the coupling equations

\[
\begin{align*}
g_{ca_x}(t) &= (x_{21} + 0.5 \cdot l_{21} \cdot \cos \varphi_{21}) - (x_{22} - 0.5 \cdot l_{22} \cdot \cos \varphi_{22}) = 0, \\
g_{ca_y}(t) &= (y_{21} + 0.5 \cdot l_{21} \cdot \sin \varphi_{21}) - (y_{22} - 0.5 \cdot l_{22} \cdot \sin \varphi_{22}) = 0, \\
g_{cp}(t) &= \varphi_{21} - \varphi_{22} = 0
\end{align*}
\]

for the fixed joint. The co-simulation has been carried out with different macro-step sizes, namely \(H = 1E - 3, H = 5E - 4\) and \(H = 1E - 4\).

Figure 5. Planar four-bar linkage: Interpretation as two double pendulums coupled by a fixed joint.
Figure 6. Simulation results for the four-bar linkage for different macro-step sizes: Displacements $x_{21}(t), y_{21}(t)$ and angle $\phi_{21}(t)$, Lagrange multipliers $\lambda_{ca_x}(t), \lambda_{ca_y}(t)$ and $\lambda_{cp}(t)$, constraint equations $g_{ca_x}(t), g_{ca_y}(t)$ and $g_{cp}(t)$. 
3 CONCLUSIONS

An analytical analysis of the numerical stability of an index-2 co-simulation approach has been presented. The analysis is based on a co-simulation test model, which can be interpreted as two algebraically coupled Dahlquist equations. In contrast to numerical time integration schemes, where the numerical stability can be calculated as a function of 2 independent parameters, 5 independent parameters are necessary to characterize the numerical stability of co-simulation methods with constraint coupling. The presented co-simulation approach shows good numerical stability properties. Stable simulations can be achieved in a wide parameter range and also for the case that the subsystems have very different mechanical properties. The global errors of the position and velocity variables converge with $O(H^2)$; the corresponding local errors converge with $O(H^3)$. The coupling of 2 nonlinear mechanical subsystems by means of algebraic constraint equations has been demonstrated in a numerical example.

REFERENCES


Numerical Model For Simulating Cam Shifting Systems In Automotive Engines

Martin Busch*, Sebastian Wolf*, Jonas Förster*, Carsten Träbing*

* Technical Calculation Dynamic Simulation, Business Division Engine Systems
Schaefller Technologies AG & Co. KG,
Industriestrasse 1-3, 91074 Herzogenaurach, Germany
Martin.Busch@schaefller.com

ABSTRACT

In order to improve the efficiency of automotive engines, a so-called cam shifting system can be applied. Thereto, the valve train is enhanced by a mechanical unit, the cam piece, which switches the lift of the valve according to the drivers demands. The switching between up to three different cams is accomplished by means of a pin-groove coupling. In the current paper a multibody model for simulating cam shifting systems is presented. The model is formulated as a nonlinear DAE system. The corresponding numerical aspects are discussed in detail. Finally, the simulation results are compared to measurements.

1 INTRODUCTION

One of the central topics in the automotive industry is the reduction of emission and fuel consumption. The improvement of the engine efficiency, including the same or even better engine performance established over the past years, is a challenging task which cannot be faced with standard approaches. A promising concept to solve the requirement is “variability”, i.e. variable engine components which ensure a thermodynamically optimized system for different load situations and engine speeds. One of these variability approaches is Schaefflers cam shifting system (CSS) where each engine valve train is driven by two or three cams mounted on a so-called cam piece, see Fig.1. By an axial movement of the cam piece in base circle phase the switching between the cams is performed. The cams are tailored to the customer demands, e.g. for achieving a cylinder deactivation in order to reduce the emission in city traffic, for achieving more engine power in a passing maneuver or for cruising on a highway with constant velocity.

An important aspect of the CSS is the switching between the cams which is typically achieved by means of a mechanical pin-groove coupling, i.e. a pin enters a pre-designed groove on the cam piece and forces the cam piece to move into another working position, see Fig.1. The contact forces between the pin and the groove contour are crucial for the durability of the pin and the safety of the system. Since the cam piece rotates very fast, only a small time window is available for the switch. Hence, steep contour shapes may have to be applied which can entail high contact forces. Also the friction is relevant for the functioning of the CSS. If the friction is too high, the cam piece may come to rest in an inconvenient position during the switch which can result in unnecessary wear of the components.

Since the system depends on multiple parameters, e.g. the engine speed or the oil pressure in the hydraulic lash adjuster, it is characterized by a complex behavior. For the design of a proper working CSS numerical simulations are necessary. Therefore, a multidisciplinary dynamical model will be presented in this paper.

In section 2 the multibody model of the CSS is introduced in detail. Section 3 deals with numerical aspects and simplification approaches. Further, an approach is presented to determine the unknown friction parameters in the model. In section 4 the simulation results of the model are discussed and compared to measurements. It is shown that the crucial phenomena are covered by the model.

2 MODEL OF THE CAM SHIFTING SYSTEM

The CSS is implemented as a multidisciplinary multibody model which is described in the following. The basic operation principle is rather simple, see Fig.1. By means of an actuator which is fixed to the inertial system, a pin can be inserted into the groove on the cam piece. The groove is designed with a
prescribed switching contour to perform a specific shift of the cam piece. Due to the pin-contour contact the cam piece is shifted axially on the camshaft until the next cam lobe is located over the finger follower. Hence, the intake and outtake cam contour of the valve train can be defined with the axial position of the cam piece on the shaft.

2.1 Replacement model for the CSS

The mechanical part of the CSS is modeled in a simplified manner. Topologically, the system can be regarded as a nonlinear 2-mass oscillator with mass $m_{\text{cam}}$ for the cam piece and $m_{\text{pin}}$ for the pin, see Fig. 2. The cam piece is modeled as a rigid body with one degree of freedom describing the axial translational motion $x_{\text{cam}}$ on the rotor shaft with respect to the inertial-fixed actuator. The angular velocity of the cam piece is assumed to be constant, so the rotation angle $\varphi_{\text{cam}}(t)$ is defined rheonomically as a linear function of time. The rotation angle defines the axial position $x_{\text{contour}}$ of the contour described in body-fixed coordinates on the cam piece. The contour function $x_{\text{contour}}$ is based on a cubic spline interpolation where the sampling points are free parameters in the model.

Several frictional contacts act at the cam piece, introducing the summarized applied force $f_{\text{sum}}$ into the system, e.g. forces due to the contact with the roller finger follower or due to the contact with the arrestor which attaches the cam piece axially to the camshaft, see the next section. The contact between the switching groove and the pin is handled in a special manner because it is an unilateral contact [2], see Sect. 2.3.

The pin is modeled as partially elastic. The bending motion of the pin tip within the actuator is influenced by the elastic stiffness. In order to achieve a realistic behavior of the pin, a nonlinear spring is assumed between pin tip and top. Force-displacement measurements are obtained from an INSTRON test rig which are averaged and fitted with a nonlinear regression function, see Fig. 3 (left). The regression function describing the axial force is defined as
with the coefficients $a_i$ and $b_i$ which are determined by the measured data. As can be seen in the figure, the nonlinear fitting function regards a steeper force gradient for large displacements $x$ which gives more realistic results than common linear approaches, e.g. FE-reduction approaches [9]. In a similar manner the stiffness in lateral direction is considered in the model, so that the pin entails two degrees of freedom.

### 2.2 Frictional contacts at the cam piece

Several contact and friction forces are applied at the cam piece, due to contacts with surrounding parts. Since the contacts are assumed to be permanent and rigid, the different parts can be cut free and the contact forces can be determined explicitly. For the corresponding friction forces a simple Coulomb approach is applied using constant friction values.

**Arrester contact for axial locking of cam piece against shaft:**

In order to prohibit an arbitrary axial motion of the cam piece, the cam piece is attached to the shaft by means of an arrester. The arrester consists of a ball which is supported against the camshaft by means of a spring, see Fig.1. The arrester ball engages with a notch milled into the inner side of the cam piece, fixing the cam piece on the shaft axially. If the CSS switches, the spring force is overcome and the ball enters another notch, arresting the cam piece again.

Cutting the arrester ball free, see Fig.3 (right), the force balance can be calculated. Since the dynamic forces of the arrester ball are neglected, the contact force in normal direction acting at the cam piece is determined by

$$
\lambda_A = \frac{f_{spr}(\Delta s)}{(1 - \mu_A^2) \cdot \cos \alpha + 2 \cdot \mu_A \cdot \sin \alpha \cdot \text{sign}(\dot{x}_{cam})}, \quad \Delta s = \Delta s(x_{cam}), \quad \alpha = \alpha(x_{cam})
$$

(2)
with \( \text{sign}(\cdot) \) the sign of the axial cam velocity and \( f_{spr} \) the spring force depending on the spring deflection \( \Delta s \). The deflection and the slope angle \( \alpha \) of the notch define the profile of the notch and are both given as spline functions. The friction value \( \mu_A \) is a user defined parameter.

Projecting the contact force into the axial direction and summing it up with the corresponding friction force yields the resulting axial arrestor force

\[
f_{arre} = \lambda_A \cdot \sin \alpha - \lambda_A \cdot \mu_A \cdot \cos \alpha \cdot \text{sign}(x_{\text{cam}}). \tag{3}
\]

It should be noted that the resulting forces in radial direction \( F_{arre,\text{radial}} \) as well as the resulting torques \( M_{arre} \) with respect to a reference position on the cam piece are also considered in the model, both are defined as 3-dimensional vectors. For sake of a clear representation, these forces/torques are not discussed in detail here and are summarized in the 6-dimensional vector

\[
\Phi_{arre} = \begin{pmatrix} R_x(-\varphi_{arre}) \cdot F_{arre,\text{radial}} \\ R_x(-\varphi_{arre}) \cdot M_{arre} \end{pmatrix}, \quad R_x(\alpha) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{pmatrix}. \tag{4}
\]

Since the axial forces are already regarded in \( f_{arre} \), the first component in \( \Phi_{arre} \) is zero. Also note that the forces/torques are defined in the arrestor-fixed coordinate system rotating with the camshaft. For reasons explained later in the current section, the forces/torques are transformed to the body fixed system of the cam piece - which is also rotating with the shaft - by means of a rotation matrix \( R_x \) using a constant angle \( \varphi_{arre} \).

**Contact to finger follower incorporating hydraulics in lash adjuster:**

A further contact occurs between the cam on the cam piece and the roller finger follower (RFF) in the valve train. The RFF is pushed against the cam by means of a hydraulic lash adjuster in order to avoid clearance in the contact. The normal force from the contact can be calculated explicitly as

\[
\lambda_{ff} = f_{\text{preload}} + \frac{P_{\text{hlal}} \cdot D_{\text{piston}}^2}{4} \tag{5}
\]

with \( f_{\text{preload}} \) the preload of the hydraulic lash adjuster spring, \( P_{\text{hlal}}(\cdot) \) a characteristic chart function describing the pressure in the hydraulic lash adjuster which depends on temperature, oil viscosity and
other parameters and \( D_{\text{piston}} \) the diameter of the piston in the HLA. It should be noted that the CSS can only switch axially when the valve is closed. In this state the lash adjuster spring force is zero. Further, the inertia forces of the valve and the RFF can be neglected because the system is in quasi-static state. Additionally, valve spring vibrations in base circle phase are neglected. Since the contact forces only act in radial direction of the cam piece, the axial resulting force of the contact consists only of the corresponding friction force given by

\[
f_{\text{rff}} = \lambda_{\text{rff}} \cdot \mu_R \cdot \text{sign}(x_{\text{cam}})
\]  

(6)

where \( \mu_R \) denotes the Coulomb friction coefficient. As already done for the arrestor approach, the radial forces and the resulting torques are collected in a 6-dimensional vector

\[
\begin{pmatrix}
R_x(-\varphi_{\text{rff}} + \varphi_{\text{cam}}(t)) \cdot E_{\text{rff, radial}} \\
R_y(-\varphi_{\text{rff}} + \varphi_{\text{cam}}(t)) \cdot M_{\text{rff}}
\end{pmatrix}
\]  

(7)

depending on \( \lambda_{\text{rff}} \). The first component is zero again. The forces/torques are transformed from the body fixed frame of the RFF into the body-fixed frame of the cam piece by means of a constant angle \( \varphi_{\text{rff}} \) and the time dependent cam angle \( \varphi_{\text{cam}}(t) \). The latter angle is needed because the RFF is a world-fixed part and does not rotate with the camshaft.

**Redundant contact in camshaft spline for radial bearing of cam piece:**

The cam piece is mounted on the camshaft by a plain bearing. The camshaft surface contains a spline profile in order to transfer the rotational motion to the cam piece. Hence, the contact between cam piece and shaft is redundant. A sketch of the bearing is shown in Fig.4. As can be seen, the contact is distributed in different normal directions \( e_i \) \((i = 1 \cdots m)\) where \( m \) denotes the amount of contact edges considered in the spline profile. Each direction entails a positive contact force \( \lambda_i \geq 0 \) since only compressive forces can contribute to the contact. The summarized contact force in radial direction of the cam piece can be formulated in matrix-vector form

\[
E_{\text{shaft, normal}} = \lambda_1 \cdot e_1 + \cdots + \lambda_m \cdot e_m = (e_1, \cdots, e_m) \cdot \lambda_{\text{shaft}}
\]  

(8)

where the normal contact forces are collected in a vector \( \lambda_{\text{shaft}} \). The resulting torques can be formulated according to

\[
\]
\[ M_{\text{shaft,normal}} = l_1 \times \lambda_1 \cdot \mathbf{e}_1 + \cdots + l_m \times \lambda_m \cdot \mathbf{e}_m = (l_1 \times \mathbf{e}_1 \quad \cdots \quad l_m \times \mathbf{e}_m) \cdot \hat{\lambda}_{\text{shaft}} \]  

where the vectors \( r_j \) denote the lever arms from the reference frame on the cam piece to the contact points. Assuming that \( \hat{\lambda}_{\text{shaft}} \) is already calculated, the axial resulting force consisting of the friction force in the plain bearing can be given in scalar form as

\[ f_{\text{shaft}} = \mu_S \cdot \lambda_1 \cdot \text{sign}(x_{\text{cam}}) + \cdots + \mu_S \cdot \lambda_m \cdot \text{sign}(x_{\text{cam}}) = \mu_S \cdot \text{sign}(x_{\text{cam}}) \cdot \hat{\lambda}_{\text{shaft}} \]  

with \( \mu_S \) the user-defined friction coefficient. The friction torques with respect to the reference frame on the cam piece are given in vector form as

\[ M_{\text{shaft,friction}} = \mu_S \cdot \lambda_1 \cdot \text{sign}(x_{\text{cam}}) \cdot l_1 \times \mathbf{e}_x + \cdots + \mu_S \cdot \lambda_m \cdot \text{sign}(x_{\text{cam}}) \cdot l_m \times \mathbf{e}_x \]

\[ = \mu_S \cdot \text{sign}(x_{\text{cam}}) \cdot (l_1 \times \mathbf{e}_x \quad \cdots \quad l_m \times \mathbf{e}_x) \cdot \hat{\lambda}_{\text{shaft}} \]  

where \( \mathbf{e}_x \) denotes the unit vector in axial direction. The radial forces and the resulting torques can be summarized in a 6-dimensional vector

\[ \Phi_{\text{shaft}} = \begin{pmatrix} E_{\text{shaft,normal}} \\ M_{\text{shaft,friction}} \end{pmatrix} = \begin{pmatrix} \mathbf{e}_1 & \cdots & \mathbf{e}_m \\ l_1 \times \mathbf{e}_1 & \cdots & l_m \times \mathbf{e}_m \end{pmatrix} + \mu_S \cdot \text{sign}(x_{\text{cam}}) \begin{pmatrix} 0 \\ l_1 \times \mathbf{e}_x \quad \cdots \quad l_m \times \mathbf{e}_x \end{pmatrix} \cdot \hat{\lambda}_{\text{shaft}} \]

Again, the first component of the vector is zero. It should be noted, that the vectors \( \mathbf{e}_j \) and \( r_j \) are defined in the body-fixed frame of the cam piece. Hence, the matrices \( E_1 \) and \( E_2 \) are constant and can be calculated in a preprocessing step. Furthermore, an additional transformation of the forces/torques into the cam-piece system, as it was necessary for the arrestor and the RFF, is not needed here.

The numerical solution of \( \hat{\lambda}_{\text{shaft}} \) will be discussed in the numerical section 3.

### 2.3 Unilateral pin/groove contact

A special aspect in the system is the contact between the switching contour on the cam piece and the pin which actually accelerates the cam piece in axial direction. Since the pin can leave the contour, the contact results in an unilateral constraint which is derived in the following.

In order to describe the contact behavior, the surface of the cam piece is unwound in cylinder coordinates, see Fig.5 (left). The cylinder coordinates are represented by a 3-dimensional tuple \((x, \phi, r)\) where the corresponding Cartesian coordinates are defined as

\[ r = T(x, \phi, r) := \begin{pmatrix} x \\ r \cos \phi \\ r \sin \phi \end{pmatrix} \]

In the figure the axial direction \( x \) is depicted over the circumferential direction \( \phi \), the radial component \( r \) is not shown. The figure shows a section of the contour expressed in the body fixed frame \( N \) of the cam piece. The coordinates of the pin \( P \) and the cam piece \( N \) are defined with respect to the centric inertial frame at the actuator position \( O \). The pin \( P \) engages the contour in a contact point \( C \) so that a contact force is applied at \( C \) in the normal direction of the contour.

By means of transformation (13) the position of the pin \( P \) can be expressed with respect to the centric cam-piece frame \( N \) resulting in the vector

\[ \]
\[ N_{\text{pin}} = T(x_{\text{pin}} - x_{\text{cam}}, \varphi_{\text{pin}} - \varphi_{\text{cam}(t)}, R_{\text{cam}}) \]  

(14)

where the radial component is assumed to be the radius of the cam piece \( R_{\text{cam}} \). The position of the potential contact point \( C \) with respect to the centric frame \( N \) is defined as

\[ N_{\text{contact}} = T(x_{\text{contour}}, \varphi_{\text{contact}}, R_{\text{cam}}) \]  

(15)

where \( \varphi_{\text{contact}} \) denotes the relative angle between \( C \) and \( N \). The scalar function \( x_{\text{contour}} \) describes the contour in axial direction, as mentioned in Sect. 2.1. The tangential vector \( t \), the normal vector \( n \) and the radial vector \( b \) of the contact point are defined according to an orthonormal Frenet system

\[
N_t = \frac{d}{d\varphi_{\text{contact}}} N_{\text{contact}}, \quad N_b = \begin{bmatrix} 0 \\ \cos(\varphi_{\text{contact}}) \\ \sin(\varphi_{\text{contact}}) \end{bmatrix}, \quad N_n = \frac{t \times b}{\| t \times b \|}
\]  

(16)

depending on the derivative of the contour function with respect to the angle \( \varphi_{\text{contact}} \) of the contact point.

If pin \( P \) contacts the contour, a constraint has to be satisfied in order to prevent interpenetration in normal direction. Treating the contact and the non-contact case in one equation, the corresponding unilateral scalar constraint reads as

\[
N_n^T \cdot (N_{\text{contact}} - N_{\text{pin}}) \leq 0.
\]  

(17)

Since constraint (17) depends on the unknown angle \( \varphi_{\text{contact}} \), the contact point \( C \) is not determined. Hence a contact-finding condition for calculating \( \varphi_{\text{contact}} \) has to be applied. With the assumption that only continuous contours are considered for the switching, the circumferential contact position can be defined according to

\[
\varphi_{\text{cam}(t)} + \varphi_{\text{contact}} = \varphi_{\text{pin}}
\]  

(18)

---

\(^1\) Note that a shift in the \( \varphi \) component of the cylinder coordinates by an angle \( \alpha \) is equivalent to the transformation of the Cartesian coordinates with a corresponding rotational matrix \( R_\alpha(\varphi) \), see e.g. Eq. (4).
which means that the potential contact point $C$ is always located under the pin in circumferential direction even if the contact is open, see Fig. 5 (right). Inserting (18) into (17), the contact point $C$ is determined and the unilateral constraint simplifies to the axial component

$$x_{pin} \geq x_{cam} + x_{contour} (\varphi)$$  \hspace{1cm} (19)$$

Due to the constraint equation a contact force acts at the pin and at the cam piece in normal direction, prohibiting interpenetration between both parts. This force is denoted by the scalar value $\lambda_{uc}$ and is determined implicitly by the constraint (19) in the contact case while it is zero in the non-contact case.

Additionally, a friction force acts in tangential direction. By summing up both forces and projecting the result into axial direction with the axial unit vector $\mathbf{e}_x = (1, 0, 0)^T$, the resulting axial force for the unilateral contact is obtained as

$$f_{uc} = \mathbf{e}_x^T \cdot N_{\text{contact}} \cdot \lambda_{uc} + \mathbf{e}_x^T \cdot N_{\text{t}} \cdot \mu_{uc} \cdot \lambda_{uc} \cdot \text{sign}(x_{cam} - x_{pin})$$  \hspace{1cm} (20)$$

with the friction coefficient $\mu_{uc}$ in the contact. The sign of the contact velocity in tangential direction is replaced by the sign of the relative axial velocity between pin and cam. Again, the resulting forces in radial direction as well as the resulting torques are collected in a 6-dimensional vector. The torque vector with respect to the cam-piece system $N$ can be calculated as

$$M_{uc} = N_{\text{contact}} \times \left( N_{\text{N}} \cdot \lambda_{uc} + N_{\text{t}} \cdot \mu_{uc} \cdot \lambda_{uc} \cdot \text{sign}(x_{cam} - x_{pin}) \right)$$  \hspace{1cm} (21)$$

Similarly, the radial forces $F_{uc, radial}$ can be calculated and the 6-dimensional resulting vector is defined as

$$\Phi_{uc} = \begin{pmatrix} F_{uc, radial} \\ M_{uc} \end{pmatrix}$$  \hspace{1cm} (22)$$

with a zero-valued first component. Since the forces/torques are already represented in the body-fixed frame $N$ of the cam piece, a further transformation as shown in Eq. (4) is not necessary.

It should be remarked that the contact position normally varies in radial direction due to the tilting of the pin in the actuator and due to the bending of the pin. This radial variation is neglected in the model, resulting in a constant radial component $R_{cam}$. Further, a second contact occurs between pin and cam piece at the bottom of the groove. This contact is not considered in the current model.

### 2.4 Equations of motion

Collecting all equations from above, the system of equations of motion for the model can be summarized in the following DAE

$$\begin{align*}
x_{cam} &= v_{cam}, \quad \dot{x}_{pin} = v_{pin}, \quad y_{pin} = w_{pin} \quad \text{(kinematic ODEs)} \\
m_{cam} \ddot{x}_{cam} &= f_{arre} + f_{shaft} + f_{ff} + f_{uc} (\lambda_{uc}) \quad \text{(dynamic axial ODE for cam)} \\
 &= f_{res} (x_{cam}, v_{cam}, x_{pin}, v_{pin}, \lambda_{shaft}, \lambda_{uc}) \\
m_{pin} \ddot{x}_{pin} &= f_{stiff} (x_{pin}) + d_{struct} v_{pin} - f_{uc} (\lambda_{uc}) \quad \text{(dynamic axial ODE for pin)} \\
m_{pin} \ddot{y}_{pin} &= f_{stiff} (y_{pin}) + d_{struct} w_{pin} - h_{uc} (\lambda_{uc}) \quad \text{(dynamic lateral ODE for pin)} \\
x_{pin} &\geq x_{cam} + x_{contour} (\varphi_{pin} - \varphi_{cam} (t)) \quad \text{(unilateral contact condition)} \\
0 &= Q_{arre} + Q_{shaft} + Q_{ff} + Q_{uc} (\lambda_{uc}) \quad \text{(static force/torque vectors)} \\
 &= \Phi_{res} (x_{cam}, \lambda_{cam}, \lambda_{shaft}, \lambda_{uc}) \\
y_{pin} &= R_{cam} \cdot \varphi_{pin}, \quad \dot{y}_{pin} = R_{cam} \cdot \dot{\varphi}_{pin}
\end{align*}$$  \hspace{1cm} (23)$$
with $f_{res}$ summarizing all axial forces acting at the cam piece and $\phi_{res}$ summarizing the static force/torque balances in the remaining directions. The constant value $d_{struct}$ denotes the structure damping in the pin. The pin has a second degree of freedom $y_{pin}$ in the lateral direction. Therefore, the contact force in lateral direction $h_{uc}$ is calculated in a similar manner as for the axial direction, see Eq. (20). Further, a transition between the translational lateral coordinate $y_{pin}$ and the angle in circumferential direction $\varphi_{pin}$ is needed. This transition is approximated by a simple rolling condition $y_{pin} \approx R_{cam} \cdot \varphi_{pin}$, since $y_{pin}$ and $\varphi_{pin}$ are small and oscillating around zero.

3 NUMERICAL SOLUTION OF THE EQUATIONS OF MOTION

The equation system (23) shall be solved numerically by means of standard time-integration methods. Therefore, the equations have to be handled in a special manner which is discussed in the following.

Handling of unilateral constraint:

As a first problem, numerical standard methods cannot deal with inequalities and hence with unilateral constraints. For incorporating inequalities into differential equation systems, different approaches exist which can be used for solving Eq. (23), e.g. methods for linear complementarity problems (LCP) [1][3][6] or proximal point formulations [4]. In the current paper the inequality is handled by means of a simple case differentiation which is also a very common approach in real-time applications (see e.g. the methods using collision detection techniques [5]). By evaluating Eq. (19) the contact case can be separated from the non-contact case, resulting in different types of equations and different contact forces. In the non-contact case the contact force $\lambda_{uc}$ is zero and Eq. (23) can be expressed as an ODE system which is solvable with standard methods. In the contact case, inequality (19) changes to an equality

\[ x_{pin} = x_{cam} + x_{contour} (\varphi_{pin} - \varphi_{cam}(t)) \]  

(24)

Determining the contact force $\lambda_{uc}$ implicitly and Eq. (23) results in a DAE system.

Reduction of DAE index:

A second numerical problem occurs in the contact case. The arising DAE system has a differentiation index of 3 which can entail instability problems in the numerical time integration of the model, see Ref. [10]. Hence, the index has to be reduced. As a first approach for the reduction, the pin is considered as static so that the inertia forces $m_{pin} \ddot{x}_{pin}$ and $m_{pin} \ddot{y}_{pin}$ in axial and lateral direction are zero and the equations for the pin simplify to

\[ 0 = f_{stiff}(x_{pin}) + d_{struct} \nu_{pin} - f_{uc}(\lambda_{uc}), \quad \text{(dynamic axial ODE for pin)} \]

\[ 0 = f_{stiff}(y_{pin}) + d_{struct} w_{pin} - h_{uc}(\lambda_{uc}), \quad \text{(dynamic lateral ODE for pin)} \]  

(25)

This simplification can be motivated from a physical point of view since the bending motion in the pin is normally very small. Numerically, it is also advantageous because the high-frequent oscillations of the pin are avoided in the system. Hence, a larger time step size can be used by the solver which results in a faster simulation.

Considering the pin as static, the index of Eq. (23) is reduced to 2. A further reduction to index 1 is achieved by differentiating constraint (24) with respect to time which results in the hidden constraint on velocity level

\[ \dot{x}_{pin} = \dot{x}_{cam} + (\varphi_{pin} - \varphi_{cam}(t)) \cdot \frac{d}{d\varphi} x_{contour} (\varphi_{pin} - \varphi_{cam}(t)) \]  

(26)
Substituting the axial pin position and the axial pin velocity in Eq. (25) by (24) and (26), the axial equation in (25) can be solved algebraically for the contact force $\lambda_{uc}$. Inserting $\lambda_{uc}$ into Eq. (23), the equations of motion become an explicit index-1 DAE system

**contact case:**

\[
\begin{align*}
\dot{x}_{cam} &= v_{cam}, \\
\dot{y}_{pin} &= h(x_{cam}, v_{cam}, y_{pin}), \\
m_{cam} \ddot{x}_{cam} &= f_{arre} + f_{shaft} + f_{ff} + f_{uc}(g(x_{cam}, v_{cam}, y_{pin})) \\
&= \hat{f}_{res}(x_{cam}, v_{cam}, y_{pin}, \lambda_{shaft}), \\
\lambda_{uc} &= g(x_{cam}, v_{cam}, y_{pin}), \\
0 &= \Phi_{arre} + \Phi_{shaft} + \Phi_{ff} + \Phi_{uc}(g(x_{cam}, v_{cam}, y_{pin})) \\
&= \hat{\Phi}_{res}(x_{cam}, v_{cam}, y_{pin}, \lambda_{shaft})
\end{align*}
\]

**non-contact case:**

\[
\begin{align*}
\dot{x}_{cam} &= v_{cam}, \\
m_{cam} \ddot{x}_{cam} &= f_{arre} + f_{shaft} + f_{ff} + f_{uc}(0) \\
&= \hat{f}_{res}(x_{cam}, v_{cam}, \lambda_{shaft}), \\
\lambda_{uc} &= 0, \\
0 &= \Phi_{arre} + \Phi_{shaft} + \Phi_{ff} + \Phi_{uc}(0) \\
&= \hat{\Phi}_{res}(x_{cam}, v_{cam}, \lambda_{shaft})
\end{align*}
\]

with new summarizing functions $\hat{f}_{res}, \hat{\Phi}_{res}$ in the contact case and $\hat{f}_{res}, \hat{\Phi}_{res}$ in the non-contact case.

The contact force $\lambda_{uc}$ is explicitly computable from a summarizing function $g$. The kinematic ODE for the lateral pin motion is substituted by a new first-order ODE with a right-hand side $h$. In the non-contact case the pin motion is not required for computing the motion of the cam piece, hence, the pin equations are skipped.

System (27) consists only of ODEs and the two explicit equations for the contact force $\lambda_{uc}$ and the normal forces $\lambda_{shaft}$ in the shaft contact. Therefore, the system can be solved by standard ODE methods in both cases [10].

**Calculation of redundant contact forces:**

A third numerical problem is related with the computation of the normal forces $\lambda_{shaft}$ in the shaft contact. Summing up all static forces/torques in the non-axial directions, results in the equation system $0 = \Phi_{arre} + \Phi_{shaft} + \Phi_{ff} + \Phi_{uc}$ in equation (23). Substituting $\Phi_{shaft}$ by the definition (12), yields a linear equation system for $\lambda_{shaft}$

\[
\begin{bmatrix}
E_1 + \mu S \cdot \text{sign}(\dot{x}_{cam}) \cdot E_2
\end{bmatrix} \cdot \lambda_{shaft} = -\Phi_{arre} - \Phi_{ff} - \Phi_{uc}.
\]

with a zero-valued first row. Eq. (28) is underdetermined, yielding 5 equations for the $m \gg 5$ unknown normal forces in $\lambda_{shaft}$. Since a unique solution does not exist, a least square solution is calculated instead:

\[
\lambda_{shaft} = \max \left( \left[ E_1 + \mu S \cdot \text{sign}(\dot{x}_{cam}) \cdot E_2 \right]^+ \cdot (-\Phi_{arre} - \Phi_{ff} - \Phi_{uc}), 0 \right)
\]

where $\left[ E_1 + \mu S \cdot \text{sign}(\dot{x}_{cam}) \cdot E_2 \right]^+$ denotes the Moore-Penrose pseudo inverse of the equation matrix, see Ref. [10]. Equation (29) yields a particular solution of the system with a minimal Euclidian
norm $\| \lambda_{\text{sh}} \|_2 = \min$ and minimal normal forces in the shaft contacts. The negative components in $\lambda_{\text{sh}}$ are ignored, since only compressive normal forces can contribute to the contact. This is accomplished in Eq. (29) by means of a component-wise maximum function, comparing each force with zero.

It should be remarked that the equation matrix in (28) is state dependent. However, the matrix and its pseudoinverse can be pre-calculated for the two cases $\text{sign}(\dot{x}_{\text{cam}}) > 0$ and $\text{sign}(\dot{x}_{\text{cam}}) < 0$ in order to save calculation effort.

### 3.1 Calibration of friction and damping parameters

A common problem with contact-dominated models is that friction is hard to estimate. Complex friction model approaches exist where the microscopic surface structure in the contact is taken into account, see e.g. [7][8]. However, such methods suffer from large computational effort. For that reason, in our implementation a simple Coulomb friction approach is applied as explained in Sect. 2. The unknown friction parameters $\mu_A$, $\mu_R$, $\mu_S$ and $\mu_{uc}$ as well as the structure damping $d_{\text{struct}}$ in the pin are determined with a parameter optimization approach. Therefore, the model is calibrated to a cam shifting system analyzed on a test rig. The simulation results are compared to the measurements in a least square sense, and the unknown parameters are optimized within a pre-defined amount of iterations in order to achieve best conformability. Local or global approaches are usable for the optimization, e.g. downhill-simplex methods or differential evolution.

The calculation time for an optimization is relatively moderate taking approximately one day to improve the starting solution sufficiently. If the parameters are determined once, they are fixed in the model until a new fitting is necessary. The subsequent simulations are computable a lot faster, taking only seconds or minutes which is a benefit of the simplicity of the model.

### 4 SIMULATION RESULTS

Solving the model by means of the standard MATLAB integrator ODE45 yields results as shown in Fig.6. The figure compares the simulation (red curve) with measurements (blue curve) for one cam rotation. As can be seen, the results fit very well and the model works with adequate correctness.

On the left side of the figure, the trajectory of the pin is plotted relative to the cam piece, i.e. the observer sits on the moving cam piece and watches the pin. The groove contour consisting of two flanks is shown as dashed line. The contour as well as the pin motion are represented in unwound cylinder coordinates. On the right side of the figure, the axial contact force $f_{uc}$ is depicted.

As can be seen, the behavior of the model is quite nonlinear. The cam piece starts in arrested state with the arrestor ball located in the middle of the current arrestor notch, see also Fig.1. At a cam angle of approx. 190° the axial switch of the cam piece is initiated, resulting in a peak in the axial force because the pin contacts the switching contour. Then, the cam piece is guided at the pin in the range 190°-250°. In this range, the axial contact force firstly rises and then decays, because the contact force is counteracted by the arrestor force which is mainly determined by the nonlinear profile in the arrestor notch. At 250° the pin leaves the contour and the contact force drops. Afterwards the cam piece moves forward due to the inertia force. If the arrestor ball overcomes the top of the current arrestor notch, the ball slides into the next notch, see also Fig.1 and Fig.3 (right), and the cam piece is pulled towards the arrestor position on the shaft. Hence, the cam piece is accelerated strongly within the interval 250°-252°. At 252° the pin is contacting the opposite contour flank in the switching groove which results in a high peak in the contact forces. The cam piece is then guided again at the pin and the contact forces decay. If the arrestor ball reaches the middle of the new arrestor notch, the cam piece is fixed again in the arrested state and the switch is finished.

There are still some minor differences between the measurement and the simulation, e.g. at the beginning of the switching. On the one hand, these differences result from the fact that the measurement
Fig. 6: Comparison of measurement (blue) and simulation (red). On the left side the trajectory of the pin midpoint in the unwound cam piece groove is plotted. On the right side, the corresponding axial contact force is plotted.

curves are averaged over multiple switches. On the other hand, the offset can result from physical effects which are not covered by the current model, e.g., misalignment of the roller finger follower. If the peaks in the contact force between cam piece and pin are too high, the pin can be damaged. In order to avoid too high forces, the switching contour and the model parameters have to be designed in an appropriate manner by the user. Improving the forces for a wide range of operating points can be a challenging task since the forces tend to increase for higher rotor speeds, for instance. In order to automate and accelerate this design process, improved model parameters as well as the sampling points of the contour spline function can be provided by the optimization approach mentioned in section 3.1.

5 CONCLUSIONS

A nonlinear model for simulating cam shifting systems (CSS) is presented. Firstly, the problem is formulated as index-3 DAE system and is then transformed into an explicit index-1 DAE system which is solvable in a similar manner than the corresponding ODE. Hence, common numerical time-integration methods can be applied to the model in order to achieve an efficient simulation. Even though the model is rather simple, the transient behavior is quite complex which is caused by several frictional contacts and clearance in the system. The comparison of the simulation with measurements yields very good results. It can be stated that the main physical effects in the CSS are covered by the model, e.g., resting of the cam piece due to large friction, double contacts at the contour due to rebound or lifting of the cam piece from the contour due to inertia effects.

REFERENCES


On the Stability of Explicit and Implicit Co-Simulation Approaches

Pu Li*, Daixing Lu*, Bernhard Schweizer*

*Institute of Applied Dynamics
Technical University of Darmstadt
Otto-Berndt-Strasse 2, 64287 Darmstadt, Germany
[li, lu, schweizer]@sds.tu-darmstadt.de

ABSTRACT
Applying a co-simulation approach, the numerical stability of the coupling method is a crucial point. In this work, we consider explicit and implicit co-simulation approaches, where the two subsystems are assumed to be coupled by constitutive equations, i.e. we consider the case that the subsystems are connected by applied forces/torques. This manuscript can be regarded as a follow-up work with respect to Ref. [7]. While in Ref. [7] results have only been presented for the symmetrical co-simulation test model, here the unsymmetrical case is discussed in detail.

Keywords: co-simulation, solver coupling, subcycling, parallelization, stability.

1. TEST MODEL FOR ANALYZING THE STABILITY OF CO-SIMULATION METHODS
The numerical stability of time-integration schemes is defined by Dahlquist’s test equation. From the mechanical point of view, this equation can be interpreted as the complex representation of the equations of motion of the autonomous linear mass-spring-damper oscillator. To analyse the numerical stability of co-simulation methods, it is therefore straightforward to use the linear two-mass oscillator as test model, see Figure 1. Obviously, this system can be interpreted as two single-mass oscillators (masses $m_1/m_2$, spring constants $c_1/c_2$, damping coefficients $d_1/d_2$), which are coupled by the coupling spring $c_c$ and the coupling damper $d_c$. In this sense, the two-mass oscillator can be regarded as two Dahlquist equations to be coupled by a linear constitutive equation.

Figure 1. Linear two-mass oscillator: Test model for analyzing the stability of co-simulation methods.

Introducing the dimensionless time $\bar{t} = \frac{t}{H}$ (H denotes the macro-step size of the co-simulation approach) and the 7 parameters

$$
\begin{align*}
\bar{c}_1 &= \frac{c_1 \cdot H^2}{m_1}, \\
\bar{d}_1 &= \frac{d_1 \cdot H}{m_1}, \\
\alpha_{m21} &= \frac{m_2}{m_1}, \\
\alpha_{c21} &= \frac{c_2}{c_1}, \\
\alpha_{d21} &= \frac{d_2}{d_1}, \\
\alpha_{cc1} &= \frac{c_c}{c_1}, \\
\alpha_{dc1} &= \frac{d_c}{d_1},
\end{align*}
$$

(1)

the equations of motion for the two-mass oscillator can be rewritten as

$$
\begin{align*}
\dot{x}_1' &= \bar{v}_1 \\
\dot{v}_1' &= -\bar{c}_1 \cdot x_1 - \bar{d}_1 \cdot \bar{v}_1 + \alpha_{cc1} \cdot \bar{c}_1 \cdot (x_2 - x_1) + \alpha_{dc1} \cdot \bar{d}_1 \cdot (\bar{v}_2 - \bar{v}_1)
\end{align*}
$$

(2)
In the above equations, \( x_1, x_2 \) denote dimensionless position variables. \((\cdot)' = \frac{d}{dt}\) terms the derivative with respect to the dimensionless time \( \tilde{t} \). \( \tilde{v}_1 = H \cdot v_1 \) and \( \tilde{v}_2 = H \cdot v_2 \) are dimensionless velocities. It should be mentioned that for \( m_1, m_2, c_1, c_2, c_c, d_1, d_2, d_c > 0 \), the two-mass oscillator is a stable mechanical system.

In Section 2, stability plots are presented for explicit and implicit co-simulation methods using different approximation polynomials (polynomial degrees \( k = 0, 1, 2 \)). All three decomposition techniques are considered, namely force/force-, force/displacement- and displacement/displacement-decomposition.

1.1 Force/Force-Decomposition Approach

Making use of a force/force-decomposition technique, the co-simulation test model is split into two subsystems so that both subsystems are force-driven single-mass oscillators, see Figure 2a. Both masses are driven by the coupling force \( \lambda_c \). The coupling force is defined by the implicit coupling equation

\[
\tilde{g}_c := \lambda_c - c_c \cdot (x_2 - x_1) - d_c \cdot (v_2 - v_1) = 0.
\]

Using dimensionless parameters, the decomposed system is described by the subsequent system of equations

**Subsystem 1:**

\[
x_1' = \tilde{v}_1 \\
\tilde{v}_1' = -\tilde{c}_1 \cdot x_1 - \tilde{d}_1 \cdot \tilde{v}_1 + \lambda_c
\] (a)

**Subsystem 2:**

\[
x_2' = \tilde{v}_2 \\
\tilde{v}_2' = -\frac{\tilde{c}_2}{\tilde{m}_2} \cdot x_2 - \frac{\tilde{d}_2}{\tilde{m}_2} \cdot \tilde{v}_2 - \frac{1}{\tilde{m}_2} \lambda_c
\] (b)

**Coupling condition:**

\[
\tilde{g}_c := \lambda_c - \alpha_{c1} \cdot \tilde{c}_1 \cdot (x_2 - x_1) - \alpha_{d1} \cdot \tilde{d}_1 \cdot (\tilde{v}_2 - \tilde{v}_1) = 0.
\] (c)

Note that \( \lambda_c = \frac{\lambda_c H^2}{m_1} \) denotes the dimensionless coupling force.
1.2 Force/Displacement-Decomposition Approach

Applying a force/displacement-decomposition technique, subsystem 1 will be a force-driven
and subsystem 2 a base-point excited single-mass oscillator, see Figure 2b. Therefore, the
coupling force \( \bar{L}_c \) is replaced in subsystem 2 with the help of the coupling condition (3c). Since
the state variables \( x_1 \) and \( v_1 \) are not available in subsystem 2, they have to be replaced by two
additional coupling variables, which are denoted by \( \bar{x}_2 \) and \( \bar{v}_2 \). Introducing two additional
coupling variables necessitates the definition of two additional coupling conditions, namely
\( g_{cx_1} := \bar{x}_1 - x_1 = 0 \) and \( g_{cv_1} := \bar{v}_1 - v_1 = 0 \).

The decomposed system is mathematically defined by the following equations of motion

\[
\begin{align*}
\text{Subsystem 1:} & \\
x'_1 &= \bar{v}_1 \\
\bar{v}'_1 &= -\bar{c}_1 \cdot x_1 - \bar{d}_1 \cdot \bar{v}_1 + \bar{L}_c \quad (a)
\end{align*}
\]

\[
\begin{align*}
\text{Subsystem 2:} & \\
x'_2 &= \bar{v}_2 \\
\bar{v}'_2 &= -\frac{\alpha_{c21}}{\alpha_{m21}} \cdot \bar{c}_1 \cdot x_2 - \frac{\alpha_{d21}}{\alpha_{m21}} \cdot \bar{d}_1 \cdot \bar{v}_2 - \frac{\alpha_{cc1}}{\alpha_{m21}} \cdot \bar{c}_1 \cdot (x_2 - \bar{x}_1) - \frac{\alpha_{dc1}}{\alpha_{m21}} \cdot \bar{d}_1 \quad (b) \\
& \quad \cdot (\bar{v}_2 - \bar{v}_1)
\end{align*}
\]

Coupling conditions:
\[
\begin{align*}
\bar{g}_{cl} & := \bar{L}_c - \alpha_{cc1} \cdot \bar{c}_1 \cdot (x_2 - x_1) - \alpha_{dc1} \cdot \bar{d}_1 \cdot (\bar{v}_2 - \bar{v}_1) = 0 \\
\bar{g}_{cx_1} & := \bar{x}_1 - x_1 = 0 \\
\bar{g}_{cv_1} & := \bar{v}_1 - v_1 = 0. \quad (c)
\end{align*}
\]

1.3 Displacement/Displacement-Decomposition Approach

Using a displacement/displacement-coupling approach, both subsystems are base-point excited
single-mass oscillators, see Figure 2c. The coupling spring/damper system is duplicated, i.e. the
coupling variable \( \bar{L}_c \) is replaced in both subsystems with the help of the coupling condition (3c).
As a consequence, 4 coupling variables and 4 coupling conditions have to be defined.

The decomposed system is characterized by the subsequent equation system

\[
\begin{align*}
\text{Subsystem 1:} & \\
x'_1 &= \bar{v}_1 \\
\bar{v}'_1 &= -\bar{c}_1 \cdot x_1 - \bar{d}_1 \cdot \bar{v}_1 + \alpha_{cc1} \cdot \bar{c}_1 \cdot (\bar{x}_2 - x_1) + \alpha_{dc1} \cdot \bar{d}_1 \cdot (\bar{v}_2 - \bar{v}_1) \quad (a)
\end{align*}
\]

\[
\begin{align*}
\text{Subsystem 2:} & \\
x'_2 &= \bar{v}_2 \\
\bar{v}'_2 &= -\frac{\alpha_{c21}}{\alpha_{m21}} \cdot \bar{c}_1 \cdot x_2 - \frac{\alpha_{d21}}{\alpha_{m21}} \cdot \bar{d}_1 \cdot \bar{v}_2 - \frac{\alpha_{cc1}}{\alpha_{m21}} \cdot \bar{c}_1 \cdot (x_2 - \bar{x}_1) - \frac{\alpha_{dc1}}{\alpha_{m21}} \cdot \bar{d}_1 \quad (b) \\
& \quad \cdot (\bar{v}_2 - \bar{v}_1)
\end{align*}
\]

Coupling conditions:
\[
\begin{align*}
\bar{g}_{cx_1} & := \bar{x}_1 - x_1 = 0 \\
\bar{g}_{cv_1} & := \bar{v}_1 - v_1 = 0 \\
\bar{g}_{cx_2} & := \bar{x}_2 - x_2 = 0 \\
\bar{g}_{cv_2} & := \bar{v}_2 - \bar{v}_2 = 0. \quad (c)
\end{align*}
\]
2. STABILITY OF EXPLICIT AND IMPLICIT CO-SIMULATION METHODS

Discretizing the decomposed co-simulation test model – i.e. Eqs. (3), (4) and (5) – with a linear co-simulation approach, a linear system of recurrence equations can be derived, which describes the time discrete solution of the co-simulation approach. The stability of the co-simulation method can simply be determined by calculating the spectral radius \( \rho \) of the recurrence system, see Ref. [7]. If the spectral radius is larger than 1, the co-simulation is called numerically unstable. The spectral radius is a function of the 7 independent test model parameters defined in Eq. (2) and depends on the polynomial degree \( k \) used for approximating the coupling variables. For characterizing the stability behaviour of co-simulation methods and in accordance with the 2D-stability plots used for illustrating the numerical stability of time integration schemes, it is convenient to use a different set of independent parameters, namely

![Stability plots: Force/force-decomposition, explicit and implicit, \( k = 0 \).](image)
Figure 4. Stability plots: Force/force-decomposition, explicit and implicit, $k = 1$.

$$\lambda_{r1} = -\frac{\ddot{d}_1}{2}, \quad \lambda_{11} = \frac{1}{2} \sqrt{4 \cdot \dot{c}_1 - \ddot{d}_1^2},$$

$$\alpha_{m21} = \frac{m_2}{m_1}, \quad \alpha_{tr21} = \frac{\ddot{d}_1}{\lambda_{r1}}, \quad \alpha_{d21} = \frac{\dot{d}_1}{m_{21}}, \quad \alpha_{l21} = \frac{1}{\lambda_{l1}} \sqrt{4 \cdot \alpha_{m21} \cdot \alpha_{c21} \cdot \dot{c}_1 - \alpha_{d21}^2 \cdot \ddot{d}_1^2} \sqrt{4 \cdot \dot{c}_1 - \ddot{d}_1^2},$$

$$\alpha_{lrc1} = \frac{\alpha_{dc1}}{\alpha^*_{m}}, \quad \alpha_{lci1} = \frac{1}{\alpha^*_{m}} \sqrt{4 \cdot \alpha^*_{m} \cdot \alpha_{c1} \cdot \dot{c}_1 - \alpha_{dc1}^2 \cdot \ddot{d}_1^2} \sqrt{4 \cdot \dot{c}_1 - \ddot{d}_1^2} \quad \text{with} \quad \alpha^*_{m} = \frac{2 - \alpha_{m21}}{1 + \alpha_{m21}}. $$
\( \hat{\lambda}_{r1} \) and \( \hat{\lambda}_{i1} \) denote the dimensionless real and imaginary part of the eigenvalue of subsystem 1. Subsystem 2 is characterized by the three parameters \( \alpha_{m21} \), \( \alpha_{rr21} \) and \( \alpha_{ri21} \), which describe the ratio of the subsystem masses as well as the ratio of the real and the imaginary part of the eigenvalue of subsystem 2 with respect to \( \hat{\lambda}_{r1} \) and \( \hat{\lambda}_{i1} \) (i.e. the ratio of the subsystem damping and the ratio of the subsystem frequencies). The coupling of the two subsystems is described by the two parameters \( \alpha_{rrc1} \) and \( \alpha_{ric1} \). In order to illustrate the stability behaviour of different coupling approaches in 2D-stability plots, it is useful to fix 5 parameters and to plot the spectral radius as a function of the remaining 2 parameters. Therefore, the parameters \( \hat{\lambda}_{r1} \) and \( \hat{\lambda}_{i1} \) are varied in the range \([-1,0]\) and \([0,2]\) for the explicit co-simulation methods and in the range \([-2,0]\) and \([0,10]\) for the implicit coupling schemes.

**Figure 5.** Stability plots: Force/force-decomposition, explicit and implicit, \( k = 2 \).
Figure 6. Stability plots: Force/displacement-decomposition, explicit and implicit, $k = 0$.

By specifying the 7 test-model parameters, the spectral radius $\rho$ of the related system of recurrence equations can be computed. It should be stressed that the spectral radius can only be calculated numerically. The circles in the 2D-stability plots indicate stable points, i.e. points for which $\rho \leq (1 + 10^{-16})$ holds. To minimize floating point errors, calculation of $\rho$ has been accomplished with 128 decimal places so that points at the boundary of the stability region ($\rho \approx 1$) are computed very precisely.
4. CONCLUSIONS

Results of the stability analysis for the explicit co-simulation methods can be summarized as follows:

- The parameter $\alpha_{m21}$ only slightly affects the numerical stability.
- Increasing $\alpha_{\lambda21}$, the numerical stability is often (slightly) improved.
- Especially for the force/force-decomposition approach, increasing the parameter $\alpha_{\lambda21}$ entails a reduced stability behaviour.
- Very crucial for the numerical stability are the parameters $\alpha_{\lambda r21}$ and $\alpha_{\lambda ic1}$. Larger values of $\alpha_{\lambda r21}$ yield a reduction of the stability region in $\tilde{\lambda}_{r1}$-direction. Larger values of $\alpha_{\lambda ic1}$ entail a reduction with respect to the $\tilde{\lambda}_{i1}$-direction.
A stable co-simulation for undamped systems is only possible with the force/force-decomposition approach using quadratic approximation functions \((k = 2)\).

For the implicit co-simulation methods, we observe:

- The best numerical stability shows the force/force-decomposition approach in combination with constant approximation polynomials \((k = 0)\).
- Good stability behaviour is also observed for the force/force-decomposition approach with \(k = 1\), for the force/displacement-decomposition approach with \(k = 1\) and for the displacement/displacement-decomposition approach with \(k = 1\).
- The parameters \(\alpha_{\text{fc1}}\) and \(\alpha_{\text{dic1}}\) have a significant influence on the numerical stability. Especially \(\alpha_{\text{dic1}}\) affects the stability with respect to the \(\lambda_{11}\)-direction.

**Figure 8.** Stability plots: Force/displacement-decomposition, explicit and implicit, \(k = 2\).
Figure 9. Stability plots: Displacement/displacement-decomposition, explicit and implicit, $k = 0$.

REFERENCES


Figure 10. Stability plots: Displacement/displacement-decomposition, explicit and implicit, $k = 1$.


Figure 11. Stability plots: Displacement/displacement-decomposition, explicit and implicit, $k = 2$.


ABSTRACT
This research describes the development of a 2 MW wind turbine model under consideration of damping and mass distribution at the tower structure. The wind turbine is a horizontal axis, pitch-controlled wind turbine characterised by a rated power of 2.05 MW and a hub height of 141 m. The multibody model is developed in SIMPACK and applies aerodynamic forces by means of interfaces for AeroDyn and Turbsim [3,6]. The prototype wind turbine is equipped with a lattice tower that is based on a finite element model and built up in Ansys. The Ansys model is generated by an APDL (Ansys Parametric Design Language) text file defined by a MATLAB script in order to enable immediately design modifications. While lattice towers are provided with numerous joints, joint patch damping should be considered in future. Therefore a test stand is introduced and used to investigate the behaviour of a single lap joint. Based on the test stand and the received results a user force element developed in SIMPACK is shown to apply the damping forces to the multibody model of the test stand.

1 INTRODUCTION
The dynamic behaviour of wind turbines with lattice towers is influenced by the damping effects of the bolted lap joints in the tower structure. Generally, damping in dynamically loaded structures is a result of internal and external effects. External damping is based on aerodynamic or impact sound effects. Considering internal damping, joint patch damping and material damping has to be distinguished. Material damping is based on friction within the material while joint patch damping is caused by friction within the contact area of joints. Compared to joint patch damping, material damping effects are in general very small [2,7]. In [1] internal damping is described as a not entirely understood problem. In elastic body models in multibody simulations damping effects are therefore often modelled in a rather simplified way. Due to the typical modal-reduced finite element formulation of a structure in multibody systems only modal damping is considered. This approach assumes uniformly distributed damping and is only valid in the case of material damping. In order to approximate the characteristics of structural damping caused by joint patches, phenomenological models like Jenkins, Masing, Iwan or Valanis were developed [4,7]. These models assume a point-shaped joint patch and can therefore be used in multi-body-simulations. In [5] the assumption of point shaped joint patch modelling is described to be acceptable if the size of the joint patch is small in relation to the shortest wavelength of propagation.

The multibody model of a wind turbine equipped with a lattice tower, here implemented in SIMPACK (Figure 1), is described in section 2 and uses a modally reduced beam model of the tower. In this approach only one flexible body describes the complex structure of the lattice tower enabling only the implementation of modal damping.
An approach to model joint patch damping in a multibody model is to build up the framework inside the multibody environment of SIMPACK. Then each beam of the framework is separately defined as an elastic body. Joint patch damping is then taken into account by means of force elements in the joints. For this purpose a model of a single lap joint is developed in section 3. The model is built up by means of the Valanis approach using experimental results obtained from a two-mass resonator.

![Figure 1. 2MW wind turbine equipped with a lattice tower.](image)

## 2 SIMULATION MODEL OF A WIND TURBINE WITH LATTICE TOWER

### 2.1 Description of the overall model

To investigate the dynamic behaviour of a lattice tower under realistic conditions, the tower structure is combined with a previously developed multibody model of a 2 MW wind turbine [8,9]. The overall simulation model is built up in SIMPACK. The multibody topology is shown in Figure 2. The SIMPACK model mainly consists of the lattice tower and the nacelle including the drivetrain. The lattice tower is described in more detail in subsection 2.2. The model of the nacelle includes the main frame and the complete drive train. The drive train consists of the rotor, the gearbox with two planetary gear stages and a spur gear stage, and the generator. The main components like blades and tower are modelled as elastic bodies by means of a modal-reduced finite element formulation.

For optimised power production the wind turbine is equipped with a controller that is integrated into the model as a direct link library (.dll). The controller of the turbine distinguishes two modes during power production, partial load and full load. In the partial load mode the rotational speed is free while the generator torque is controlled in order to maximize the power production. Is the rated power reached in accordance to the wind conditions and generator torque, the controller changes into the full load mode. Then the generator torque is kept fixed and the pitch angle of the blades is controlled in order to keep the rotational speed and therefore the power production constant. To protect the wind turbine from overload it is shut down by pitching to 90 deg when hazardous wind conditions occur.

In the multibody model the aerodynamic loads are applied by the SIMPACK interface AeroDyn (Force Element 241) that is based on the blade element momentum theory [6]. With AeroDyn only constant wind fields can be applied to the blade and tower structure of a wind turbine. To take turbulent wind conditions into account TurbSim is additionally used [2].
2.2 Model of the lattice tower

The multibody model of the lattice tower is based on a finite element formulation developed in Ansys. For visualisation of the geometry in SIMPACK the geometry is saved in Ansys by a .cdb-file. Performing a modal reduction based on Craig Bampton method, Ansys generates a superelement saved in a .sub-file. Additionally generating a .tcms-file enables SIMPACK to use informations from a recovery matrix to display deformations correctly.

To incorporate different tower topologies into the simulation, a MATLAB script was developed. The script uses an input file generating Ansys Parametric Design Language Code (APDL) necessary to define a finite element model in Ansys.

The principle of the topology definition is shown in Figure 3. The topology is limited to four-legged lattice towers. Each node within the structure has a unique number due to a special nodal convention seen in Table 1. The script enables the user to build up a lattice tower as shown in Figure 3 with corner bars, horizontal-, diamond-, k- and additional stiffening bars on the tower faces. Variants of internal bracing definitions are shown in Figure 3c.

Figure 2. Multibody model of the 2 MW wind turbine equipped with a lattice tower. a. Overall model. b Multibody topology.

Figure 3. a Nodal definitions of corner bars and tower faces. b Horizontal-, diamond- and additional stiffening bars on the faces. c Internal stiffening bars and spider web
Table 1. Nodal convention

<table>
<thead>
<tr>
<th>Nodes at:</th>
<th>Numbering</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cornerbars</td>
<td>1, 100, 101, ..., 199; 2, 200, 201, ..., 299; 3, 300, 301, ..., 399; 4, 400, 401, ..., 499</td>
</tr>
<tr>
<td>Area middle nodes</td>
<td>500, 501, ..., 599; 600, 601, ..., 699; 700, 701, ..., 799; 800, 801, ..., 899</td>
</tr>
<tr>
<td>Middle nodes</td>
<td>9, 900, 901, ..., 999</td>
</tr>
<tr>
<td>Subarea middle nodes</td>
<td>1100, 1101, ..., 1199; 1200, 1201, ..., 1299; 1300, 1301, ..., 1399; 1400, 1401, ..., 1499; 1500, 1501, ..., 1500; 1600, 1601, ..., 1699; 1700, 1701, ..., 1799; 1800, 1801, ..., 1899</td>
</tr>
</tbody>
</table>

The finite element model consists of beam elements. In accordance to the available data the user can choose between a simple circular cross section of the bars where only the cross section area is necessary or arbitrary shaped cross sections defined by input files. For modal reduction purposes the definition of master nodes is possible at the symmetry line of the lattice tower. The master nodes at the bottom and top of the lattice tower are defined by a spider web. Moreover additional masses can be applied at corner bars, the middle nodes of each surface or at the symmetry line to consider additional masses like ladders, cables and bolts.

2.3 Simulation results

A first evaluation of the overall simulation model is the investigation of the dynamic properties represented for example by the eigenmodes. Figure 4 illustrates the first tower mode in wind direction showing the interaction of the structure components like tower and blades.

Figure 4. Visualization of the first mode in wind direction.

To investigate the dynamic behaviour under consideration of damping specific load cases defined by classification societies like the Germanischer Lloyd (GL) are advantageous. Here the Design Load Case (DLC) 2.2 is considered. It describes an operating case of a wind turbine working under normal conditions until a pitch error occurs. Figure 5 shows the simulated time
histories of the tower top displacement in wind direction and the pitch angle. The wind turbine is exposed to a normal wind profile model at an average wind speed of 13 m/s at hub height and operates under normal conditions with its rated rotor speed 1150 rpm and rated power of 2.05 MW. At \( t = 50 \) s an error occurs and the plant controller changes the pitch angle to 0 deg. As a result the generator speed runs up until 1380 rpm. After that the safety program shuts down the turbine by changing the pitch angle to 90 deg to avoid damage. Pitching to 90 deg rapidly reduces the aerodynamic drag forces causing oscillations of the structure. The decay curve of the entire structure in Figure 5a is influenced by its damping behaviour. Modal damping is applied for the tower model. The results for damping values of 1 % and 7 % show the influence and importance of damping.

![Figure 5](image)

**Figure 5.** DLC of a pitch error. a Lattice tower top displacement in wind direction. b Pitch angle.

### 3 INVESTIGATING A LAP JOINT

Within the multibody model an individual lap joint is to be modelled by a nonlinear force element. The dissipative effect of a force element is shown under dynamic load by a hysteresis of the joint force over the displacement. A typical experimental setup to determine such a hysteresis is a two-mass resonator described in the following. Subsequently a multibody model of the resonator setup is built up in order to parametrise the force element formulated as a Valanis model. A comparison of experimental and simulation results concludes this section.

#### 3.1 Test stand for damping analysis of a single bolted joint

A two-mass resonator as described in [4,5] is a suitable tool to analyse the damping characteristics of a bolted joint in tangential direction experimentally. The resonator shown in Figure 6 developed in [10,11] consists of two lumped masses suspended by thin steel wire ropes at its centres of gravity. The masses are bolted together by a lap joint using a M16 bolt. The test stand is harmonically excited by means of a shaker transferred by a stringer. To achieve the resonance effect a flexure spring is provided between mass 1 and the lap joint. A force sensor is used between stringer and mass 1 to measure the actual excitation force.

The idea of the test principle is to excite the setup with its first resonance frequency in order to increase the dynamic loads on the lap joint. The accelerations on both sides of the joint are measured with the sensors 8. The tangential force acting on the lap joint is calculated with the measured acceleration of mass 2 as the inertia force

\[
F_t = -m_2 \alpha_2
\]

The displacements \( u_1 \) and \( u_2 \) of the masses 1 and 2 are obtained by numerical integration of the accelerations up to the position level in accordance to the initial conditions. The relative tangential displacement of the two parts connected by the lap joint then is obtained as

\[
\Delta u = u_2 - u_1.
\]
While the measurement results contain noise the received data must be treated first [5]. Using band-pass-filter before integrating the raw acceleration data, results can be enhanced.

To verify the resonance effect of the test setup, it was excited with a force sweep. Experimental modal analysis showed the first tangential mode at 558 Hz. The design is adjusted that the first eigenmode of the resonator seen in Figure 7b is a mode in longitudinal direction only.

![Figure 6](image1.png)  
**Figure 6.** Two-mass resonator for analysing damping properties of a lap joint [10,11].

The primary purpose of the design in [10,11] was to realize a high-strength friction grip fastening in accordance to DIN EN 14399 and to permit the entire deformation cone in the contact area due to VDI 2230 (Figure 7a).

![Figure 7](image2.png)  
**Figure 7.** a Developed stress contribution due to high-strength friction grip fastening [10].  
  b First mode of the test design at 558 Hz [10].

Dynamic excitation of the system at its resonance shows hysteresis loops for the tangential force $F_t$ over the displacement $\Delta u$. The hysteresis shown in Figure 8 for one cycle was obtained for a tightening torque of the bolt of less than 2 Nm. The hysteresis is here caused both by micro- and macroslip in the lap joint. Under realistic conditions only microslip effects should occur.
3.2 Simulation model of the resonator

Topology

The simulation model of the two-mass resonator is built up with SIMPACK. Masses 1 and 2 are modeled as elastic bodies based on finite element formulation developed in Ansys and reduced to five and three master nodes, respectively, along the main axis of the resonator (Figure 9). The interface between the FE model and the discrete force element at the lap joint is realized by spider webs, thus beam elements with the properties of high stiffness and low density in comparison to the investigated structure. The spider webs connect the nodes 100000 on both sides of the lap joint with the inner nodes of the corresponding borehole.

In the multibody model, mass 1 is connected by a 6 DOF joint with the global reference frame. Mass 2 is connected with mass 1 by a prismatic joint in tangential direction. Gravity is set to zero. The force element for the tangential lap contact is defined between the nodes 100000. The contact force law is formulated as a Valanis element [4] and implemented as a SIMPACK user element. The multibody model is harmonically excited at its first eigenfrequency using a force element acting in z-direction.

Valanis force element

The dissipative characteristic of the lap joint is modelled by a Valanis force element [4] and implemented in SIMPACK as a user force element. It enables to consider both macro- and microslip effects. The parameters of the Valanis model have to be determined experimentally. While the procedure is described in detail in [4], only a brief summary is given here.
The Valanis model leads to a differential equation relating the first-order time derivative of the tangential force $F_t$ with the tangential force itself and the displacement $\Delta u$ and its time derivative $\Delta \dot{u}$,

$$\dot{F}_t = f(\Delta u, \Delta \dot{u}, F_t).$$  \hspace{1cm} (1)$$

The differential equation to integrate within the user force element is

$$0 = \frac{k_0 \Delta \dot{u}}{1 + \kappa} \left( 1 + \frac{\Delta \dot{u}}{k_0} \right) (k_i \Delta u - F_t) \right)$$

$$\dot{F}_t = \frac{k_0 \Delta \dot{u}}{1 + \kappa} \left( 1 + \frac{\Delta \dot{u}}{k_0} \right) (k_i \Delta u - F_t)$$

where $\lambda$ is defined as

$$\lambda = \frac{k_0}{F_H \left( 1 - \kappa \frac{k_i}{k_0} \right)}$$

The meaning of the unknown force element parameters $\kappa, k_0, k_i$ and $F_H$ is shown in Figure 10a. These parameters have to be experimentally identified.

Equation (2) is derived by introducing the so-called intrinsic time $z$ in order to take the deformation history into account what is needed to model a hysteresis behavior [13].

$$d\varepsilon(t) = \left| d\dot{\varepsilon}(t) - \kappa \frac{d\sigma(t)}{E_0} \right|$$

In Eq. (4) $\varepsilon_0$ represents the current strain while the fraction term is the strain if the deformation is linear elastic with Young's modulus $E_0$. The value $\kappa$ is an adjustment factor in the range $0 \leq \kappa < 1$. Therefore $z$ is the difference between the actual strain and the strain that would result from a linear-elastic behavior. The relationship between stress and strain is defined by a differential equation in the form of the Pointing-Thomson-Model

$$\dot{\sigma} + a_1 \sigma = a_2 \dot{\varepsilon} + a_3 \varepsilon$$

\hspace{1cm} (5)
with the material parameters $a_1, a_2$ and $a_3$. It is possible to define equation (5) as a function of the intrinsic time $z$,

$$\sigma'(z) + \lambda \sigma'(z) = E \varepsilon'(z) + \lambda E \varepsilon(z).$$

(6)

Transferring Eq. (6) to the discrete force in the lap joint results in

$$F'_i(z) + \lambda F_i(z) = k_0 \Delta u_i(z) + \lambda k \Delta u(z).$$

(7)

Transforming equation (7) and under consideration of the intrinsic time the time derivation of the tangential force acting in the contact area is calculated as described in equation (2).

### 3.3 Comparison of measurement and simulation

In SIMPACK the differential equation (2) is embedded into a user routine and integrated by the explicit Euler method. During the simulation the user element returns at each time step tangential forces and applies them to the lap joint.

To evaluate the user routine the parameters for a specific load case was identified. The load case is defined by a tightening torque of 2 Nm at the test stand. The amplitude of the harmonic excitation force is 108.6 N at a frequency of 558 Hz. The measurement time is 2.5 s. Applying a band pass filter with a range between 150 Hz and 2000 Hz the hysteresis shown Figure 8 is obtained. From this the parameters of the Valanis model are determined as follows:

$$k_0 = 1.4 \cdot 10^8 \text{ N/m}; \quad k_i = 1.2 \cdot 10^7 \text{ N/m}; \quad F_H = 56 \text{ N}.$$  

(10)

A comparison with the multibody simulation using these parameters is shown in Figure 10b. The hysteresis curves show a good compliance in its size and ascending properties. It is visible that the enclosed area of the hysteresis received in the simulation is slightly greater indicating higher damping in the simulation.

### 4 CONCLUSIONS

In order to perform multibody simulations of a wind turbine this paper describes how to build up a modal-reduced parametric model of a lattice tower. The advantages of the FE formulation for building up one superelement for the complete tower using a MATLAB script are demonstrated. The simulation of a specific Design Load Case with different modal damping parameters illustrates the influence of damping. While material damping can be adequately considered by modal damping, joint patch damping cannot be adequately addressed. In future joint patch damping could be considered by modelling the lattice tower within a multibody simulation environment. Such a model will consist of numerous individual elements connected to each other by joints and finally using force elements to consider joint patch damping. In general joint patch damping occurs in a contact area. The Valanis model enables to consider the lap joint as a point element formulation with only a few parameters that have to be determined experimentally. As a first approach a user force element was applied. Using a test stand to determine the hysteresis and the unknown parameters the force element has been validated.
REFERENCES


Variable communication steps and stiffness computation in co-simulation of large MBD and FEA systems

Isamu Ishikawa*, Motoharu Tateishi#, José L. Ortiz†

* MSC Software Corporation
Shinjuku First West 8F, Tokyo, Japan
Isamu.ishikawa@mscsoftware.com

# MSC Software Corporation
Shinjuku First West 8F, Tokyo, Japan
Motoharu.tateishi@mscsoftware.com

† MSC Software Corporation
201 Depot Street, Suite 100, Ann Arbor, MI 48104, USA
Jose.ortiz@mscsoftware.com

ABSTRACT

A co-simulation algorithm between a multibody dynamics system (MBD) and a finite element system (FEA) is presented. The algorithm implements a variable communication step between the co-simulating processes. In addition to the force and position data being exchanged, a stiffness matrix is computed by the FEA system and shared with the MBD system. The stiffness matrix is used by the MBD system to better predict the co-simulation results. A theoretical presentation and numerical examples are included.

Keywords: Co-simulation, multibody, finite elements, stiffness matrix.

1 INTRODUCTION

Most of the existing co-simulation algorithms (e.g. 1D systems) offer several options to control the accuracy and stability of the solution. However, larger FEA systems co-simulating with large MBD systems are usually limited to the exchange of force and position data. This paper presents a co-simulation algorithm in which each FEA system additionally computes a stiffness matrix and shares it with the MBD system. The stiffness matrix is a condensed global stiffness matrix involving only the degrees of freedom being controlled by the MBD system. The MBD system uses the stiffness matrix to better predict the motion of the coupled system.

The proposed algorithm is an enhanced version of that presented by Elliot [2]; it implements a variable communication interval but it enforces the MBD system to always step first. Both the MBD and the FEA systems solve their respective equations using their best settings. The MBD system may need to take more than one step if its current time step is smaller than the FEA system’s time step. All codes constantly communicate with a controller glue code (master code).

The inclusion of the stiffness matrix into the MBD system permits better predicting the co-simulation results and allows larger time step. The methodology is comparable to recent Jacobian-based co-simulation algorithms [4, 5]. However, the stiffness matrix does not take into account inertial effects during a dynamic co-simulation. During static and quasi-static simulations, the stiffness matrix becomes an exact Jacobian for the FEA system.

The paper discusses the overall implementation, limitations and industrial examples.

2 ALGORITHM DESCRIPTION

The algorithm can be described as follows. Figure 1 shows the co-simulation between an MBD model and the FEA model at an arbitrary time t. For simplicity assume the current simulation time is the same for both subsystems. The algorithm is based on the concept of interaction points. We define interaction points as the locations where two systems interact. The implemented algorithm assumes the interaction points are fixed joints between the co-
simulating systems. Other type of interaction points (e.g. revolute joints) need to be modelled in one of the interacting sub-system. The algorithm forces the MBD system to always step first.

2.1 Interaction forces and stiffness computation

The MBD system queries all forces and torques at the interactions points while prescribing both positions and rotations on the FEA system. Given that the MBD system steps first, it queries the FEA system for current value of the forces \( F_{i0} \) an \( F_{j0} \) at the interaction points as well as the tangent stiffness \( K_{ij} \). Vectors \( U_{i0} \) and \( U_{j0} \) are the current global positions at the interaction points at the start of the time step.

With the above information in hand, the MBD system assembles forces \( F_i \) and \( F_j \) as shown in Figure 2 using the following representation:

\[
\begin{bmatrix}
F_i \\
F_j
\end{bmatrix} = \begin{bmatrix}
F_{i0} \\
F_{j0}
\end{bmatrix} - \begin{bmatrix}
K_{ij}
\end{bmatrix} \begin{bmatrix}
U_i - U_{i0} \\
U_j - U_{j0}
\end{bmatrix}.
\]

(1)

Vectors \( U_i \) and \( U_j \) represent the current simulation’s global position at the interaction points. After taking the new integration step, the MBD system will impose a prescribed motion onto the FEA system and so forth.

It can be demonstrated that the configuration computed by the MBD system using this approach is equivalent to assembling a super-element with stiffness \( K_{ij} \) and mass equal to zero as shown in Figure 3.
One additional feature is the option to add a structural damping coefficient $\beta$ into the force equations:

$$\begin{bmatrix} F_i \\ F_j \end{bmatrix} = \begin{bmatrix} F_{i0} \\ F_{j0} \end{bmatrix} - \begin{bmatrix} K_i \end{bmatrix} \begin{bmatrix} U_i - U_{i0} \\ U_j - U_{j0} \end{bmatrix} - \beta \begin{bmatrix} K_j \end{bmatrix} \begin{bmatrix} V_i \\ V_j \end{bmatrix}. \quad (2)$$

The inclusion of the stiffness matrix in the interaction force computation permits the co-simulation to have more stable results and take larger time steps in both co-simulating codes.

### 2.2 Variable communication algorithm

The presented algorithm does not impose a schedule for the communication interval. Regardless of the relative stiffness of both co-simulating systems, the MBD system always advances first. If the current MBD’s time step is smaller than the current FEA’s time step, the MBD system will take additional time steps up to the point when the FEA system can take a simulation step. This case is illustrated in Figure 4 (a) where the MBD system takes steps labelled 1, 2, 3, and 4 before the FEA can take step 1. Conversely, if the FEA’s time step is smaller than the MBD’s time step, the FEA system will take several time steps up to the current simulation time in the MBD system. This case is illustrated in Figure 4 (b) where the FEA system takes steps labelled 1, 2, 3, 4 and 5 after the MBD code takes step 1.

![Figure 4. Variable communication between co-simulating codes.](image)

The communication of the MBD and FEA codes with the master code is constant; the MBD passes kinematic data while the FEA code passes force information at the end of every time step. The master code keeps a history of all data and provides extrapolated or interpolated values to each process when needed. The master code implements quadratic, linear and least
squares algorithms to compute interpolated/extrapolated data. Kinematic data and force data may be interpolated/extrapolated using different approaches.

One advantage of the implementation is that each analyst can tune the models using his expertise on the corresponding field. However, we found that 95% or more of the CPU time needed to complete the co-simulation is spent by the FEA code; hence we usually keep the MBD system running with a maximum time step close to that of the FEA’s.

2.3 Limitations and future development

The algorithm enforces extrapolated force data into the MBD system; the extrapolated force data is used to compute the kinematic data to be imposed into the FEA system. The FEA system computes the force using the imposed kinematic data. The main limitation of the algorithm is that it does not implement corrective measures when discrepancies between computed force values versus extrapolated force values are encountered. The authors are considering implementing an automatic corrective measure based on multi-rate integration in the future; see for example Gear [6].

The computation of the stiffness matrix is an additional chore to be completed by the FEA system. However, the extra computation time needed to complete the co-simulation allows taking larger time steps resulting in an overall faster co-simulation. However, given the potentially large number of values required to be stored in the history of values, no interpolation/extrapolation is performed with the stiffness matrix and the MBD code uses the latest available value. Numerical examples show this limitation has no detrimental effects in structural and thermo-structural problems solved by the FEA system.

3 NUMERICAL EXAMPLES

The proposed algorithm has been implemented in MSC Adams to co-simulate with MSC Marc. Extensive validation was performed running simple to complex co-simulation models.

3.1 All-terrain vehicle (ATV)

Figure 5 below shows an all-terrain vehicle before and after a sideways curb impact. The objective of the co-simulation was to study the damage in one lower control arm.

![Figure 5. ATV vehicle before and after a sideways curb impact.](image)

The vehicle is modelled in MSC Adams except for the lower left control arm and attachment bushings which are modelled in MSC Marc (see Figure 6.) All inertia parts in the MSC Adams model are rigid parts. The impact velocity is 12 Km/h.
The MSC Marc model consists of shell elements with an elasto-plastic steel with yield stress of 350 MPa. A dynamic solution was performed using a time step of 5.0E-5 sec in MSC Adams while a transient non-linear large-displacement simulation was performed in MSC Marc. The co-simulation is stable and the forces on the control arm are high enough to cause plastic deformation with local buckling and permanent deformations (see Figure 7).

Figure 7. Plastic deformation in lower control arm.

Figure 8 shows a chart showing the force magnitude computed by Marc at each attachment bushing. The impact with the curve happens approximately at time $t=0.33$ sec.

A similar validation was performed using proprietary models showing good results.

### 3.2 Full vehicle lane change maneuver

Figure 9 shows a standard full vehicle modelled in MSC Adams except for two non-linear bushings that are modelled in MSC Marc. Figure 10 shows details of the bushing models. The bushings are modelled using hexahedral elements made of and incompressible rubber with visco-elastic and elasto-plastic material properties.
The co-simulation consisted of a static simulation followed by a straight driving at 60 Km/h and a final lane change. The simulation run for 2 sec using 200 time steps.

![Figure 9. Full vehicle with two non-linear bushings.](image)

![Figure 10. Bushing model.](image)

![Figure 11. Bushing stresses.](image)

### 4 CONCLUSIONS

The variable communication implementation along with the computation of the stiffness matrix has been used with success in solving diverse types of MBD models co-simulating with an FEA code. Using the stiffness matrix provides more stable and faster co-simulations.
REFERENCES


Virtual sensing on mechatronic drivetrains using multiphysical models

Bart Forrier, Frank Naets, Wim Desmet

Department of Mechanical Engineering
KU Leuven
Celestijnenlaan 300B, B-3001 Heverlee (Leuven), Belgium
bart.forrier@kuleuven.be

ABSTRACT
This paper discusses coupled input/state estimation on a typical mechatronic driveline, consisting of an electric motor and a cardan transmission with flexible intermediate shaft. States and an a-priori unknown load torque are estimated using an Unscented Kalman Filter. This approach enriches the nonlinear lumped-parameter coupled electro-mechanic drivetrain model by measurement data. The measurement data - subject to realistic noise levels - is generated by a reference model which consists of a 1D motor model in combination with a 3D flexible multibody model of the cardan transmission. By employing current measurements in combination with rotational acceleration data, stable load torque estimation is numerically validated for dynamic excitation.

Keywords: Kalman filter, Input estimation, Mechatronic.

1 INTRODUCTION
In recent years, great advancements have been made with respect to mechanical system modelling. Research with respect to flexible multibody (FMB) system modelling has opened up the possibility to accurately calculate high-dynamic structural responses in terms of displacements and even strains and stresses for general mechanisms [1]. Thanks to recent advancements in the fields of model order reduction [2], such results can even be computed in real-time [3] for systems which can be described by a moderate number of relevant flexible deformation shapes. The integration of flexible multibody models in a multiphysical system model featuring coupling between electrical and mechanical dynamics also leads to insightful results. One example is insight in energy flows resulting from high-dynamic loading conditions in a weaving-machine [4]. Another example is insight in critical loading (i.e. torque reversals) of a wind-turbine’s main shaft due to the coupling between mechanical and electrical dynamics [5].

Although such high-fidelity models perform a crucial role in the design of high-performance mechanical or multiphysical systems, their full potential is currently not exploited. The main limitation is that although mechanical and system-level behaviour are described in a lot of detail, still a high level of uncertainty exists on operational loading predictions. One important source of uncertainty is the fact that operational loads are often not well-known, meaning that design engineers only have a vague idea about which inputs should be applied to the model. A second contribution to uncertainty on computed results is formed by often quite big uncertainties on model parameter values. These are often not known a-priori or feature a complex (unmodeled) dependency on e.g. temperature or state of wear.

Recently, some attention has gone to combining high-fidelity models with state estimation techniques which have a rich tradition in the field of navigation and control [6]. For these applications, low-cost estimation algorithms like the linear, extended and unscented Kalman Filter [16] (KF, EKF, UKF, see e.g. [7]) have been very successful [8]. Although most high-fidelity FMB models are not straightforward to implement in the recursive one-step predictor-corrector approach of the Kalman Filter, e.g. due to the presence of constraints [9], some recent examples of successful applications exist. Joint state/input estimation in civil engineering applications and based on a limited number of acceleration measurements is demonstrated in [10]. In [11], an EKF is employed to successfully apply combined state/input/parameter estimation on structural applications.
The work presented in this paper aims to make an advancement towards combined input/state estimation on mechatronic drivetrains. Such drivetrains - as found in e.g. wind turbines, electric vehicles and a whole range of industrial machinery - are subject to ever more stringent demands in terms of noise, vibration & harshness (NVH), energy-efficiency, and high-dynamic system behavior. Although performance of such systems has certainly increased over the last years (see e.g. [18]), a lot of problems still occur (see e.g. [12]) due to lack of detailed knowledge on operational system behaviour. This is due to a combination of aforementioned model uncertainties and the difficulty with which relevant dynamic quantities, like forces, torques or strain fields over a driveline can - or even cannot - be measured.

By proposing an approach to combine feasible-to-obtain measurement data from different domains, with a multiphysical drivetrain model, this paper shows how enriching models with measurements helps to gain information on operational system behavior. The paper demonstrates how a lumped-parameter drivetrain model can be used to simultaneously estimate an external load torque, and indicates how a high-fidelity flexible multibody model will enlarge the scope and increase accuracy of estimation results.

2 PROPOSED APPROACH

This work uses two multiphysical models of varying complexity to numerically validate coupled state/input observer results on mechatronic drivetrains. Figure 1 illustrates this approach.

First, a reference model (a) of the mechatronic drivetrain is composed. The drivetrain consists of an asynchronous induction motor, connected to a load through a double cardan transmission in Z-configuration. The load consists of an inertia on which an external load torque acts. The reference model features both torsion and bending behaviour of the intermediate shaft in the double cardan transmission.

Next, a filter model (b) of the same drivetrain is derived. This model should be fit for use in the proposed state/input observer, and allow for correct estimation of the a-priori unknown load torque. As discussed further on, the resulting filter model is highly nonlinear.

Finally, a Sigma Point Kalman Filter scheme [6] is implemented to allow fusing the nonlinear filter model with measurement data. The measurement data is generated from the reference model, after which realistic values for white noise are added. Comparison of the estimation results - specifically of the load torque - with the reference counterparts shows that the 1D model is suitable for observing torsional dynamics.

Figure 1: Proposed approach to validate estimation results using a 1D-3D reference model.
3 DRIVETRAIN MODEL

3.1 Reference model

The reference model consists of a 1D induction motor model, combined with a 3D flexible multibody model of a double cardan configuration. LMS Imagine.Lab AmeSim is used to solve both models in a co-simulation approach, employing LMS Virtual.Lab Motion to simulate the dynamic behaviour of the double cardan transmission. As the 3D flexible multibody model features system-level nonlinearity and distributed flexibility, it allows an informative assessment of the proposed estimation approach using a 1D observer model and enables to identify the added value of flexible multibody models in state estimation on mechatronic drivetrains.

The multibody model assumes the connections between yokes and cardan spiders to be ideal. The intermediate shaft is modeled as a flexible body, introducing both torsional and bending oscillations in the driveline. Bending of the intermediate shaft changes the deflection angle of each cardan joint in a different way, so that the deflection angles are no longer identical. Torsion of the intermediate shaft means that both cardan joints no longer define one common plane, so that the in- and outgoing shaft no longer have the same speed, even in case of equal deflection angles. Finally, the intermediate shaft has a periodical speed variation with respect to the in- or outgoing shaft at non-zero deflection angles. This also leads to torsional oscillations, as its inertia is greater than zero.

3.1.1 Lumped-parameter induction motor model

The induction motor model is a lumped-parameter description (see e.g. [13]). This model is readily available in AMESim’s library of Electric Motors and Drives models. It represents exactly the same physical behavior as the induction motor model used by the observer (section 4). As is common in literature, it also makes use of Park’s transformation [14] to express both stator and rotor quantities in one common rotating reference frame. The frame consists of a direct (d) and quadrature (q) axis. The homopolar (0)-component is also computed, meaning applied stator voltages do not necessarily form a balanced set. Magnetic saturation and hysteresis effects are not modeled, hence both stator and rotor inductances are assumed constant in this qd0-frame.

3.1.2 Flexible multibody model of the mechanical driveline

To simulate the behavior of the mechanical driveline in LMS Virtual.Lab Motion, a floating frame of reference component mode synthesis method (FFR-CMS) is used [1]. The intermediate shaft is modeled as a flexible body, and its corresponding set of component modes is computed using the Craig-Bampton method [15]. This method describes component flexibility by a combination of dynamic mode shapes representing the component deformation, and static modes representing rigid-body motion. Practically, the modeshape is computed based on a finite element model. This is constructed in a preprocessing phase, using MSC Patran/Nastran as mesher/solver. Linear tetra-elements are used to mesh the relatively complex shape of the yoke at each shaft end, whereas 100 linear beam elements are used for the shaft itself. The mesh is shown in figure 2, and the first five resulting modeshapes are listed in table 1. The high-dynamic response is dominated by deformation according to these modes.

Figure 3 shows the resulting flexible multibody model. Each end (yoke) of the intermediate shaft body is connected to the corresponding joint’s spider, modeled as a rigid body, by only allowing revolution along one axis of the spider. To be able to impose this relationship, two extra nodes are added to the mesh of the intermediate shaft. Each of these nodes is located in the centre of the corresponding cardan joint, i.e. where its two rotation axes cross. By imposing a multi-point constraint between each of these nodes and a corresponding set of nodes on each side of the meshed yoke (left side of figure 2), they can be used to model each yoke as if it were indeed connected to a rigid spider. As the flange of each cardan joint is also modeled as a rigid body, it can easily be constrained to only rotate along the spider’s other axis of rotation. Both the motor and load are
Figure 2: Mesh of the intermediate shaft, using a combination of linear tetra-elements (blue) and linear beam elements (grey). Multi-point constraints (Patran RBE2-elements) are shown in green.

Table 1: First five orthogonal component modes of the intermediate shaft.

<table>
<thead>
<tr>
<th>Mode nr.</th>
<th>Description</th>
<th>frequency</th>
<th>damping</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>lateral bending 1</td>
<td>1167 Hz</td>
<td>1 %</td>
</tr>
<tr>
<td>2</td>
<td>vertical bending 1</td>
<td>1200 Hz</td>
<td>1 %</td>
</tr>
<tr>
<td>3</td>
<td>lateral bending 2</td>
<td>2925 Hz</td>
<td>1 %</td>
</tr>
<tr>
<td>4</td>
<td>vertical bending 2</td>
<td>3208 Hz</td>
<td>1 %</td>
</tr>
<tr>
<td>5</td>
<td>torsion 1</td>
<td>3291 Hz</td>
<td>1 %</td>
</tr>
</tbody>
</table>

assumed rigid as well. The motor is completely fixed to the adjacent flange, and is constrained to the ground so that it can only revolve around its own axis. The load rotates around its axis and is otherwise constrained to a rigid slider, which translates in one direction with respect to the ground. The translation distance, imposed through a kinematic driver, is constant throughout the simulation.

Figure 3: Flexible multibody model of the mechanical part of the drivetrain.

3.2 Observer model
The filter model is schematically shown in figure 1b. Like the reference model, it includes electrical dynamics of the squirrel-cage induction machine as well as mechanical dynamics of the double-cardan transmission.
In the electrical part of this model, currents \(i_{d,q}\) and voltages \(u_{d,q}\) are decomposed along the direct and quadrature axis of a reference frame rotating in synchronism with the stator voltage phasor. Additional subscripts \(s\) and \(r\) denote, respectively, stator and rotor quantities. The homopolar component is omitted, implying that the three phases of each electrical quantity, including externally applied stator voltages, form a balanced set. As in the reference model of section 3.1, saturation and hysteresis effects are not included. Following these assumptions, eq. (1) describes the electromagnetic dynamics of the induction motor in continuous state space, with input matrix \(B_{qdsr}\) and state update matrix \(A_{qdsr}\) defined in eq. (2). The electromagnetic motoring torque \(T_{em}\) is given in eq. (3), where \(p\) denotes the pole pair number.

\[
\frac{d}{dt} \begin{bmatrix} i_{qs} \\ i_{ds} \\ i_{qr} \\ i_{dr} \end{bmatrix} = A_{qdsr} \begin{bmatrix} i_{qs} \\ i_{ds} \\ i_{qr} \\ i_{dr} \end{bmatrix} + B_{qdsr} \begin{bmatrix} u_{qs} \\ u_{ds} \end{bmatrix} \tag{1}
\]

\[
B_{qdsr} \triangleq \sigma^{-1} \begin{bmatrix} L_r & 0 & 0 & 0 \\ 0 & L_r & 0 & 0 \\ -L_M & 0 & -L_M & 0 \\ 0 & -L_M & 0 & -L_M \end{bmatrix}, \quad A_{qdsr}(\omega) \triangleq \sigma^{-1} \begin{bmatrix} -r_s L_r & -\omega L_s L_r - L_M^2 (\omega - \omega_r) & r_L M & -\omega_L M L_r \\ \omega L_s L_r - L_M^2 (\omega - \omega_r) & -r_s L_r & \omega_L M L_r & r_L M \\ r_s L_r & \omega L_s L_s & -r_s L_s & \omega L_s^2 - (\omega - \omega_r) L_r L_s \\ -\omega L_s L_s & r_s L_r & -\omega L_s^2 + (\omega - \omega_r) L_r L_s & -r_s L_s \end{bmatrix} \tag{2}
\]

\[
T_{em} = pL_M (i_{qs} i_{dr} - i_{ds} i_{qr}) \tag{3}
\]

In these equations, \(\omega_r = p\omega_m\) (with \(\omega_m\) the mechanical speed) is the electrical speed of the rotor and \(\omega\) is the synchronous speed. Assuming the induction machine is coupled to an ideal grid, stator voltage amplitude and frequency are constant. This allows to treat \(\omega\) as a parameter. The leakage coefficient \(\sigma\) equals \(L_s L_r - L_M^2\), with \(L_s\) and \(L_r\) respectively denoting the total stator and rotor inductance and \(L_M\) the mutual stator-rotor inductance. Stator and rotor resistance values are respectively denoted as \(r_s\) and \(r_r\).

The mechanical part of the observer model features only torsional driveline dynamics induced by the cardan joint kinematics and by the torsional flexibility of the intermediate shaft. In contrast to the reference model, torsional oscillations do not arise as a result of speed variation of the - now massless - intermediate shaft in this model. Torsional oscillations occur as a result of flexibility of the intermediate shaft, and its interaction with the cardan joint kinematics. Like in the reference model, the idealized cardan joints exhibit neither friction or play.

The relation between the torque \(T_{is}\) in the intermediate shaft and the torque acting on the motor \(T_{cm}\) or load side \(T_{cl}\) by the cardan transmission, is then given by eq. (4). In this equation, \(\omega_{is,m}\) (\(\omega_{is,l}\)) is the speed of the intermediate shaft at the motor (load) side, and \(\omega_i\) is the speed of the load. The deflection angle of the cardan joint near the motor (load) is \(\beta_m\) (\(\beta_l\)). The torque transmitted by the intermediate shaft is given in eq. (5), where \(\theta_{is,m}\) and \(\theta_{is,l}\) are the rotation angles at both ends of the intermediate shaft. Equation 6 gives their relationship to the motor (load) angle \(\theta_m\) (\(\theta_l\)). As eq. (4) to (6) indicate, the cardan joints induce nonlinear mechanical behaviour. This even holds at equal deflection angles \(\beta\), as the intermediate shaft, although modeled as a linear spring-damper combination, behaves as if it featured a nonlinear spring characteristic.
Using eq. (3) and (4), the dynamics of the mechanical part of the state vector are described by eq. (7), where $J_m$ and $J_l$ respectively denote the inertia at the motor and load side, and $T_{load}$ is the externally applied load torque.

$$\frac{d}{dt} \begin{bmatrix} \theta_m \\ \omega_m \\ \theta_l \\ \omega_l \end{bmatrix} = \begin{bmatrix} \omega_m \\ J_m^{-1}(T_m - T_{cm}) \\ \omega_l \\ J_l^{-1}T_{cl} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ -J_l^{-1} \end{bmatrix} T_{load}$$ (7)

## 4 UKF-BASED OBSERVER

Operational system behavior and loading conditions can be obtained by combining measurement data with the model presented in the previous section. This combination is possible through the use of a state observer. A great variety in such observers exists, and a useful overview was provided by Radke and Gao [8]. In the context of virtual sensing using multiphysical models, the Kalman Filter [16] and its variants are appealing candidates, as their recursive two-step prediction-correction approach is relatively inexpensive in terms of computational cost.

The prediction essentially consists of integrating a discretized model with state vector $x$ from time $t_{k-1}$ to time $t_k$. The correction is based on new measurement data $y$ at time $t_k$, and takes into account uncertainty on both the model $f$ and the measurements. This can be done by assuming zero-mean Gaussian white noise $w_k$, with covariance matrix $Q$, on the model equations (8) and similar noise $n_k$ with covariance matrix $R$ on the measurements (eq. 9).

$$x_k = f(x_{k-1}, u_{k-1}) + w_k$$ (8)

$$y_k = h(x_k, u_k) + n_k$$ (9)

In any Kalman filter approach, the prediction step provides a-priori estimates on the states and state covariances ($\hat{x}_k^-$ and $\hat{P}_{xx,k}^-$), as well as estimates of the expected measurements and their covariances ($\hat{y}_k^-$ and $\hat{P}_{yy,k}^-$). The correction step is based on the difference between expected and actually retrieved measurements, using the estimated covariances to perform an optimal weighting between model and measurement info.

In the Unscented Kalman Filter (UKF), a-priori (−) state and covariance estimates at time $k$ are generated by passing states at time $k - 1$ through the nonlinear state update function $f$ using an Unscented Transformation (UT) [17]. The UT consists of determining a set of $2n+1$ sigma points $\chi$ around the current estimate $\hat{x}$ ($n$ is the dimension of $x$), and transforming each sigma point through the nonlinear function (eq. 10 and 11). Applying predetermined weights $W_i$ to each transformed sigma point $\chi_i^*$ ($i = 1..2n+1$), results in an a-priori estimate of the state vector’s mean value and covariance matrix (eq.12 and 13). The same UT approach yields measurement predictions (eq. 14 to 16), as well as the cross covariance $\hat{P}_{xy,k}$ between states and measurements (eq. 17). When model and measurement errors $w$ and $n$ are indeed Gaussian, correctly chosen weights $W_i$ yield exact mean and covariance estimates even though only $2n+1$ samples are used.
to non-linearly transform the uncertainty distributions. In the presented work all $W_i$ are equal to $1/(2n)$, except $W_1 = 0$.

$$
\mathbf{X} = \begin{bmatrix}
\hat{x}_{k-1} \\
\mathbf{P}_{xx,k-1}^{+} \\
\mathbf{P}_{xx,k-1}^{-} \\
\end{bmatrix}
\mathbf{X}^* = f(\mathbf{X}^*, u_{k-1})
$$

(10)

$$
\hat{x}_k = \sum_{i=1}^{2n+1} W_i \chi_i^*
$$

(12)

$$
\mathbf{P}_{xx,k}^- = \mathbf{Q} + \sum_{i=1}^{2n+1} W_i (\chi_i^* - \hat{x}_k^-)(\chi_i^* - \hat{x}_k^-)^T
$$

(13)

$$
\hat{y}_k = \sum_{i=1}^{2n+1} W_i \chi_i^*
$$

(14)

$$
\mathbf{P}_{yy,k}^- = \mathbf{R} + \sum_{i=1}^{2n+1} W_i (\chi_i^* - \hat{y}_k^-)(\chi_i^* - \hat{y}_k^-)^T
$$

(15)

$$
\mathbf{P}_{xy,k}^- = \sum_{i=1}^{2n+1} W_i (\chi_i^* - \hat{x}_k^-)(\chi_i^* - \hat{y}_k^-)^T
$$

(16)

$$
\mathbf{X} = \mathbf{P}_{xy,k}^+ \left( \mathbf{P}_{yy,k}^- \right)^{-1}
$$

(18)

$$
\hat{x}_k^+ = \hat{x}_k^- + \mathbf{K}(y_k - \hat{y}_k^-)
$$

(19)

$$
\hat{y}_k^+ = \hat{y}_k^- - \mathbf{K}\hat{y}_k^- \mathbf{K}^T
$$

(20)

In this work, as in a lot of industrial cases, knowledge on the system’s states implicitly assumes knowledge on the input, i.e. $T_{load}$ (see fig. 1b). The resulting problem of coupled state/input estimation can be solved by augmenting the state vector with unknown input $u_u$, leaving only known inputs in $u$. By lack of other knowledge, the unknown input is typically assumed to follow a random-walk, i.e. to be nominally constant ($u_{u,k} = u_{u,k-1}$), yet variable due to associated process noise $w_u$.

The resulting Augmented Unscented Kalman Filter (A-UKF), predicts the augmented state vector (defined in eq. 21) based on a discretized version of eq. (1), (7) and (22). The discretization makes use of an explicit second-order Runge-Kutta scheme with a timestep of 500 $\mu$s.

$$
\frac{d\mathbf{x}^T}{dt} = 0
$$

(22)

Electrical measurements consist of the stator currents in the dq-frame, after the transformation of the currents through inverter terminals $a, b, c$ using eq. (23). In this equation $\theta_d$ denotes the
instantaneous electrical angle of the d-axis w.r.t. the axis defined by motor coil \( a \). The same transformation is also useful for determining the known input \( \mathbf{u}^T = [u_{q}, u_{d}] \).

\[
\begin{bmatrix}
    i_{qs} \\
    i_{ds}
\end{bmatrix} = \sqrt{\frac{2}{3}} \begin{bmatrix}
    \cos(\theta_d) & \cos(\theta_d - \frac{2\pi}{3}) & \cos(\theta_d + \frac{2\pi}{3}) \\
    \sin(\theta_d) & \sin(\theta_d - \frac{2\pi}{3}) & \sin(\theta_d + \frac{2\pi}{3})
\end{bmatrix} \begin{bmatrix}
    i_a \\
    i_b \\
    i_c
\end{bmatrix}
\] (23)

Mechanical measurements include shaft angle \( \theta_m \) and speed \( \omega_m \) obtained through a 1024-p encoder at the motor side, and shaft accelerations \( \dot{\omega}_m \) and \( \ddot{\omega}_l \) at either side of the driveline. The latter can practically be obtained by a Ferraris-sensor. The measurement functions of eq. (24) feature no direct feedthrough of known inputs \( \mathbf{u} \). This holds for the mechatronic system of fig. 1b, as well as for many other applications. Note that the accelerations in \( \mathbf{h} \) depend highly nonlinearly on the states \( \mathbf{x} \).

\[
\mathbf{h}(\mathbf{x}) = [i_{qs} \ i_{ds} \ \theta_m \ \omega_l \ J_m^{-1}(T_{em} - T_{cm}) \ J_l^{-1}(T_{cl} - T_{load})]^T
\] (24)

Covariance matrices \( Q \) and \( R \) are assumed diagonal. \( Q \) is chosen such that a deviation of 10 % on the model parameters does not influence the estimation results significantly, whilst \( R \) is based on datasheets of existing sensors. Specific attention is given to the element of \( Q \) corresponding to \( T_{load} \). This parameter is tuned so that fast variations on \( T_{load} \) are properly tracked, whilst not unnecessarily increasing overall noise on the estimation results.

5 DISCUSSION OF RESULTS

To validate the estimation, first a simulation is performed on the reference model of section 3.1. This simulation has a duration of 10-seconds and consists roughly of two parts. From the start up to 2.5 seconds into the simulation, the load torque equals the motor’s nominal value, and the terminal voltages form a balanced set at nominal amplitude and frequency. This corresponds to a direct online start of the motor. After about 2.5 seconds, nominal speed is reached. During the following part, a linear sine sweep from 10 Hz to 200 Hz is superposed onto the nominal load torque (fig. 4a). The amplitude starts at 10 % of the nominal torque value, and drops linearly at a rate of 0.0395 %/Hz, reaching 2.5 % at the end of the simulation (fig. 4b).

![Figure 4: Sinusoidal load torque’s (a) frequency and (b) amplitude as a percentage of the constant (nominal) load torque.](image)

Using the filter model of section 3.2 and noised measurements from the reference simulation, the A-UKF observer of section 4 is able to jointly estimate the states and a-priori unknown dynamic load torque. Accuracy of the estimates obviously depends on the noise level for the measurements, but also on how well the filter model matches the simulated behavior of the reference model.

Figure 5a shows the estimate of the motor speed \( \dot{\omega}_m \) versus the measurement values. Figure 5b shows the same estimate versus the actual motor speed of the reference model. Comparison of both
figures reveals that the speed estimate \( \omega_m \) is much more accurate than its measured counterpart. This requires a reasonably good observer model, combined with accurate measurements of other quantities. In this case, the acceleration measurements are very accurate (figure 5c), with a signal-to-noise-ratio (SNR) of 84 dB. The dynamic load torque estimation (figure 5d) also benefits greatly from these accurate measurements. The electrical current measurements (which have a rather low SNR of 20 dB) are required for stability of this (joint state/input) observer.

Figure 5: Estimates (full lines) compared to measured or reference signal (dots) for (a,b) motor speed, (c) load acceleration and (d) load torque.

Figures 6a-6i zoom in on the results from figures 5b-5d, revealing good overall correspondence between measured and estimated values at excitation frequencies up to approximately 160 Hz. It can be noted that the model-based estimation allows to identify the load torque quite well below the first driveline resonance (figures 6a-6c), around this resonance (figures 6d-6f) and above this resonance (figures 6g-6i). The latter case is particularly challenging, as the load torque changes quite rapidly between successive time steps, while measurements are still dominated by the effect of passing by the first torsional resonance.

In order to gain further insight in the quality of estimation results, waterfall diagrams of load acceleration and load torque estimation results are shown in figures 7a and 7b, respectively. These reveal that on the whole, the rather complex and nonlinear relationship between applied inputs (i.e. load torque) and measurements (i.e. load acceleration) are well-captured by the filter model. Indeed, the load acceleration shows significant frequency content between 150 Hz to 160 Hz, which can be attributed to nonlinear behavior of the cardan transmission, and around the first torsional resonance just below 100 Hz. Both effects due to system behavior are, as expected, not visible in the load torque estimate. The load acceleration between 60 and 70 Hz, though, is reflected in the load torque estimate, which should ideally not be the case. The reason is that the filter model assumes the intermediate shaft to be massless, which is not the case in the reference model. Therefore the filter correlates the resulting component on second shaft order in measured load acceleration to an erroneous component in the estimated load torque.

This model mismatch could be remedied by adding a state-dependent equivalent inertia to both the motor and the load side, representing the effect of the intermediate shaft’s rotational inertia. However, matching the driveline’s rigid behavior around the second shaft order would result in both inertia values equalling half of the actual shaft’s inertia, whereas matching the behavior around the...
first torsional resonance would require much lower equivalent inertia values. A model mismatch like this one is consequently difficult to remedy using a lumped-parameter description, whereas making the step towards a flexible multibody description in the filter model would completely avoid the problem, and allow to describe combined torsional and bending behavior. Therefore, future research will focus on embedding a FMB model in state estimation, using the Global Modal Parametrization scheme [3] to reduce the number of states and eliminate constraint equations. The purpose-built and instrumented test setup of fig. 8 will then be used for validation.

6 CONCLUSIONS
This paper has presented combined input/state estimation on a mechatronic drivetrain. The state and load torque estimation approach makes use of Unscented Kalman Filtering, combining a lumped-parameter non-linear electro-mechanical drivetrain model with measurements of electri-
cal voltages and currents, encoder data and rotational acceleration. It was validated against a numerical model consisting of a lumped-parameter induction motor model, coupled to a flexible multibody model of a cardan transmission. In spite of significant deviations between reference and filter model, the obtained estimation accuracy allows virtual sensing of dynamic torque input covering a bandwidth of 0 to approximately 160 Hz, significantly higher than the driveline’s first resonance frequency. Finally, this work points out how moving from a lumped-parameter drivetrain model towards a 1D-3D model would improve the accuracy and enlarge the scope of the applied estimation techniques. Future research on this will be validated on an experimental setup.

7 ACKNOWLEDGEMENTS
This work benefits from the Belgian Programme on Interuniversity Attraction Poles, initiated by the Belgian Federal Science Policy Office (DYSCHO). The Fund for Scientific Research - Flanders (F.W.O.) is also gratefully acknowledged for its support. The authors also acknowledge the support of ITEA2 through the MODRIO project, and of IWT Flanders through the OPTIWIND and the MODRIO projects.

REFERENCES


A Kalman filter-based algorithm for IMU signals fusion applied to track geometry estimation

Andrés F. Hidalgo∗, Javier Ros#, José Luis Escalona∗

∗ Department of Mechanical Engineering
University of Seville
Camino de los Descubrimientos S/N,
41092 - Seville, Spain
[ahidalgo5, escalona]@us.es

# Department of Mechanical Engineering
Public University of Navarre
Pamplona, 31006 - Navarre, Spain
jros@unavarra.es

ABSTRACT
The interest of determining the track irregularities of ground vehicles resides on the study of its dynamic response. An appealing low cost solution to estimate these irregularities from in-service vehicle resides on fusing the signals that comes from inertial measurement units located at the vehicle. In this article we present a simplified scenario whit the objective to estimate the irregularities of two dimensional profiles by means of a Kalman filter algorithm. A distinctive aspect of our algorithm with respect to more classical approaches of position and orientation estimation problems is that knowing the position and orientation of the monitoring vehicle at a specific time instant do not completely solve the problem. This means that the geometry of the track profile can not be completely determined from a unique vehicle time instant position and orientation. So, various time instants coordinate positions must be stored into the state vector. This characteristic introduces a nonlinearity into the problem when the vehicle is travelling with a time varying velocity, turning the data fusion algorithm more involved. The algorithm proposed in this article takes into account this issue, showing a very good matching between the actual and the estimated values. The behaviour and robustness of filter proposed have been tested by means of four simulations, where the monitoring vehicle is advancing at constant and varying velocity and over two different smooth two dimensional track profiles.

1 INTRODUCTION
Ground vehicles undergo external excitations that come from track irregularities. The measurement of these irregularities can be of interest for many applications, as in the case of railway vehicle dynamics. Condition monitoring of track geometry with in-service vehicles by means of Inertial Measurement Units IMUs represents an interesting alternative because continuous operations are possible with a lower cost than with special recording vehicles, (Weston et. al. [1]). In spite of these advantages, dealing with the noise and drift phenomenon of Micro-Electro-Mechanical-Systems (MEMS) sensors while fusing their data implies a challenging signal processing task. The theoretical estimation problem that we present in this article is based on the frequent track geometry monitoring problem that arises in railway industry applications (Faiz and Singh [2]). Our approach consists on a simplified version of the real problem, where a two dimensional scenario is considered and only pure kinematic considerations have been taken into account. In this case, the accelerometer and gyroscope signals that come from an IMU attached to a two-wheel solid vehicle are intended to be used to reconstruct the track geometry. By knowing the position and orientation of the vehicle at each time instant, the track geometry can be easily inferred. MEMS based IMUs have been extensively used in attitude and position estimation problems and there are several works about this subject in the literature. In all cases, in order to estimate both position and orientation from IMU signals, accelerations and angular velocities measurements must be integrated. Nonlinearity effects, inaccuracy introduced by noise and gain related errors are accumulated during long time integration process, resulting in very large position and orientation errors. To overcome these issues, there is a recurred strategy that consists on including a position...
sensor to compensate the position and orientation errors that come from the Inertial Measurement System (INS), see Won, et. al. ([3]) and Alam et. al. ([4]) for instance. One of the most commonly used devices to this end are the GPS, but the small geometrical scale of the irregularities we are supposed to deal with, requires position accuracies for which low cost GPS are not appropriate. Additionally to the positioning sensing problem related to limitations imposed by high accuracies and the usage of low cost measurement devices, a particular issue of our application compared with the most typical position and orientation estimation algorithms surges. This particularity resides in the fact that besides knowing the vehicle position at each time instant, the kinematic constraint related with the fact that the rear wheel of the vehicle will reach the same position that the front one had earlier in time, it must be ensured. In order to achieve an optimal estimation of the track irregularities, the statistical properties that the Kalman filter shows with linear time invariant systems are desirable, see for instance (Gelb [5], Simon [6] and Grewal and Andrews [7]). As statistics of this algorithm propagates to future time instants considering only those properties corresponding to the immediately previous ones, various vehicle positions should be concatenated into the state vector to explicitly force the fulfilment of the kinematic constraint between the front and the rear wheel of the vehicle. Due the smoothness of the track profiles considered, to estimate when the rear wheel is going to reach the same position that the front wheel had a moment before can be easily done when the vehicle is moving at constant velocity. On the contrary, when the advance velocity of the vehicle varies on time, this implies a more involved task.

With the aim of providing a general solution to this problem, in this article we develop a Kalman filter implementation which is able to successfully estimate the track irregularities of random two dimensional profiles at time varying advance velocity.

The body of the article consist of four sections including the conclusions. Section 2 presents the Kalman filter algorithm which is proposed, Section 3 shows the simulation results obtained with two different track geometry and time velocity functions. Finally, at Section 4 the conclusions are drawn.

2 MODEL EQUATIONS

In order to obtain vertical track irregularities, the relation between the IMU signals and the vertical coordinate $h$ of the track must be established. To derive these equations it is useful to observe the Figure 1 which shows the two-dimensional scheme of the monitoring vehicle to be used. It must be taken into account that small curvatures are considered so, linearity conditions are accepted.

As shown in Figure 1, the relation between the slope of the track profile $h^0$ and the vertical coordinate change rate $\dot{h}^0(t)$ of the front axle wheel can be easily expressed as

$$\dot{h}^0(t) = \dot{s}(t) h^0$$

where $\dot{s}(t)$ is the advance velocity of the vehicle.

By taking the derivative of Equation (1) the instantaneous vertical acceleration of the IMU is
obtained
\[ h^0(t) = h^0(0) + \dot{h}^0 \dot{s}(t) + \ddot{s}(t) h^0 \]  
(2)
where \( h^0 \) is the curvature of the track.

On the other hand, the angular rate of the vehicle \( \dot{\phi} \) is related to the vertical coordinate change rate of both wheels as follows
\[ \dot{\phi} = (\ddot{h}^0(t) - \ddot{h}(t)) / L \]  
(3)
In this way, Equations (2) and (3) show the relation among the IMU signals, the track geometry and the kinematic variables of the model.

Additionally to these signals, the information of an encoder located at the front wheel is supposed to be provided. To this end, letting \( s(t=0) = 0 \), the angle rotated by this wheel at a specific time \( t \) will be
\[ \theta(t) = \frac{s^0(t)}{R} \]  
(4)
By using the information provided by the measurements, the knowledge we have of our system and the assumptions made about the system noise statistics and measurement errors, we need to estimate the state of our system at a specific time based on the last measurement data of the sensors disposed. To that end, the Kalman filter provides one of the most extensively used state estimation techniques for linear systems.

If a discrete time system evolution is supposed, \( k \) being the time index advance, the Kalman filter is described by the following equations
\[
x_k = A_{k-1}x_{k-1} + G_{k-1}u_{k-1} + w_{k-1} \\
z_k = H_k x_k + v_k
\]  
(5)
(6)
where \( x \) is the system state vector, \( u \) is a control input, \( z \) is the measurement vector and \( w \) and \( v \) represent the process and measurement noises. \( A, H \) and \( G \) are matrices with appropriate dimensions, \( A \) and \( H \) being the more noticeable and represent the state transition matrix and the measurement one.

With the objective to obtain an optimal estimation of the vertical coordinate \( h \) of the track profile by means of the previously described signals of an IMU located in the wheel axle of a vehicle advancing with a variable velocity \( \dot{s} \), these signals will be further analysed.

2.1 Process model

Designating the acceleration measured by the IMU located in the axle of the front wheel as \( \ddot{h}^{\text{measured}} \), the true acceleration of the front wheel of the vehicle \( \ddot{h}^0 \) can be decomposed by the accelerometer measurement, its bias and a Gaussian noise as follows
\[ \ddot{h}^0 = \ddot{h}^{\text{measured}} - \ddot{h}^{\text{bias}} + w^\ddot{h} \]  
(7)
where \( w^\ddot{h} \) represents the accelerometer noise.

Although other more involved models of accelerometer exist, considering that the bias represents one of the main source of errors of these sensors, the Equation (7) results good enough for our purposes.

As the accelerometer is supposed to be placed at the front wheel of the vehicle, its vertical coordinate at time \( k + 1 \) based on the acceleration at time \( k \) can be estimated by means of the basic kinematic relation
\[
h^0_{k+1} = h^0_k + \dot{h}^0_k \Delta t_s + (1/2) \Delta t_s^2 \left( \ddot{h}^{\text{measured}} - \ddot{h}^{\text{bias}} + w^\ddot{h}_k \right)
\]  
(8)
Equation (8) indicates that if \( h^0_k, \dot{h}^0_k \) and \( \ddot{h}^{\text{bias}}_k \) are considered as state variables, the measured acceleration can be considered as a control variable while \( w^\ddot{h}_k \) is white noise. The sampling rate \( \Delta t_s \) is a known parameter based on the shortest wave length irregularities of interest and the fastest
velocity to be reached. So, as the velocity of the front wheel is going to be part of the state vector, its estimate at time $k+1$ will be

$$\dot{h}_{k+1}^0 = \dot{h}_k^0 + \Delta t_s \left( \ddot{h}_{\text{measured}}^k + \ddot{h}_{\text{bias}}^k + w_k^k \right)$$  \hspace{1cm} (9)$$

As it can be seen in Equation (3), the angular velocity is directly related to the velocities of the end points of the vehicle. As the time interval $\Delta t_s$ will usually be shorter than the time that the rear wheel takes to reach the same position that the front wheel, so many samplings will occur in this time. So, in order to use Equation (3), a relation between the coordinate of the end points is required. This relation can be easily established considering that as the vehicle advances, the coordinate of the point 1 at time $k+1$ will be the coordinate that point 0 had at the previous time $k$.

$$h_k^1 = \dot{h}_k^0; \ldots; h_k^{i+1} = h_k^i; i = 0, \ldots, n - 1$$

Extending this reasoning to all the intermediate points between the end points of the vehicle, the velocity of point $L$ can be obtained as

$$\dot{s}_0 = \dot{h}_0 = \ddot{s}_0 \Delta t + \frac{1}{2} \Delta t^2 \left( \ddot{s}_{\text{control}}^k + w_k^k \right)$$  \hspace{1cm} (13)$$

In the same way, the front wheel horizontal velocity estimate at time $k+1$ will be

$$\dot{s}_0 = \dot{s}_0 + \Delta t_s \left( \ddot{s}_{\text{control}}^k + w_k^{k} \right)$$  \hspace{1cm} (14)$$

Based on the previous analysis, determined by Equations (8), (9), (10), (13) and (14) and with the aim to express our model by means of Equation (5), the state vector at time $k$ must contain not only the coordinate and velocity of the end point 0, but also the coordinates of all the intermediate points until the point $n$ as well, as shown

$$[x_k]_{N \times 1} = \begin{bmatrix} h_k^0 & s_k^0 & h_k^1 & s_k^1 & h_k^2 & s_k^2 & \ldots & h_k^n & s_k^n & \ddot{h}_{\text{bias}}^k \end{bmatrix}^T$$  \hspace{1cm} (15)$$

where $N = (n+1) + 1$

As the control variables are $\ddot{h}_{\text{measured}}^k$ and $\ddot{s}_{\text{control}}^k$, the control vector at time $k$ will be

$$u_k = \begin{bmatrix} \ddot{h}_{\text{measured}}^k & \ddot{s}_{\text{control}}^k \end{bmatrix}^T$$  \hspace{1cm} (16)$$
Then, the state transition matrix $A_k$ will have the following form

$$A_k = \begin{bmatrix}
1 & 0 & \Delta t_s & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 & -0.5 \Delta t_s^2 \\
0 & 1 & 0 & \Delta t_s & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 & -0.5 \Delta t_s^2 \\
0 & 0 & 1 & 0 & \Delta t_s & 0 & \cdots & 0 & 0 & 0 & 0 & 0 & -\Delta t_s \\
1 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & 0 & \cdots & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \cdots & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
\end{bmatrix}_{N \times N}$$

On the other hand, the matrix $G_k$ corresponding to the control term of Equation (5) will be

$$G_k = \begin{bmatrix}
0.5 \Delta t_s^2 & 0 \\
0 & 0.5 \Delta t_s^2 \\
\Delta t_s & 0 \\
0 & \Delta t_s \\
0 & 0 \\
\vdots & \vdots \\
\vdots & \vdots \\
0 & 0 \\
\end{bmatrix}_{N \times 2}$$

At this point all the terms of Equation (5) corresponding to the Kalman filter process model of a LTI system has been already defined. To complete the model definition the measurement equations remain to be established.

### 2.2 Measurement model

In Section 2.1 it was shown that the accelerometer signal was used as a control variable. In this sense, to be used as measurement signals only those coming from the encoder and gyroscope are left. Thus, with the purpose of rendering the system observable, additionally to encoder and gyroscope, a virtual sensor to measure the track profile vertical coordinate is added. This virtual sensor will always be measure zero, so its error will equate the magnitude of the vertical irregularity.

In order to obtain the complete set of equations which define the measurement model in the form of Equation (6) both, the encoder and gyroscope signal must be expressed as a function of the state space variables.

To do so, let us first consider the encoder signal, which theoretical expression is depicted by Equation (4). In this equation it is supposed that the encoder does not have any noise. This fact can be expressed in a more realistic fashion if a normal distributed noise is assigned to the data coming from this device. So, the encoder signal at time $k$ will be treated as

$$\theta_k = \left( \frac{s^0_k}{R} \right) + v_{\theta k}$$

$v_{\theta k}$ being a normal distributed encoder noise.

Regarding the gyroscope expression, combining Equations (11) and (10) with Equation (3) and taking into account a normal gyroscope noise $v_{\phi k}$, the following expression of the gyroscope signal at time $k$ as a function of the state space variables is obtained

$$\phi_k = \frac{1}{L} \left( \dot{h}_k^0 - \frac{h_k^0 - h_{k-1}^0}{s_k^0 - s_{k-1}^0} s_k^0 \right) + v_{\phi k}$$

724
Equation (20) indicates that the gyroscope has a non-linear dependency of the state space variables so, Equation (6) it would more precisely be written as

\[ z_k = f(x_k) + v_k \]  

(21)

Due to the fact that we are considering small vehicle rotations, it is expected that errors coming from linealization will be acceptable. In this way, expanding Equation (21) as a Taylor series and neglecting second and higher order terms, the measurement equation left

\[ z_k = H_k x_k + v_k = \nabla f(x_k) + v_k \]  

(22)

As non-linearity of the measurement equation comes from the gyroscope model only, the values of the random variable associated with this equation can be interpreted as the values with respect to a nominal track irregularity from which \( \dot{\phi}_k = 0 \) at any time \( k \).

With all these considerations in mind, the measurement vector at time \( k \) will be

\[ z_k = [0 \, \theta_k \, \dot{\phi}_k]^T \]  

(23)

Therefore, the measurement matrix \( H_k \) of Equation (22) will have the following form

\[
H_k = \begin{bmatrix}
1 & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots \\
0 & 1/R & 0 & \cdots & 0 & 0 & 0 & \cdots \\
0 & 0 & 1/L & 0 & \cdots & \frac{\partial \dot{\phi}}{\partial h_k} & \frac{\partial \dot{\phi}}{\partial s_k} & \cdots \\
\end{bmatrix}_{3 \times N}
\]  

(24)

where all the partial derivatives with respect to \( h_k \) and \( s_k \) with superscript different from \( l \) and \( l-1 \) are equal to zero.

Recalling from Figure 1 that \( s^l < s^L < s^{l-1} \), when the advance velocity \( \dot{s} \) is constant, once the coefficients \( l \) and \( l-1 \) are determined they will remain unchanged all the time. On the contrary, when the advance velocity \( \dot{s} \) is variable these coefficients can change from time to time so, the measurement matrix \( H_k \) must be updated at each filter propagation time.

3 SIMULATION RESULTS

In this section, the simulation results obtained by applying the described Kalman filter algorithm to estimate the vertical irregularity coordinate of two different track profiles are presented. In all the cases, a simulation time of 100 s was chosen.

The first track profile consists of a sinusoidal irregularity with an amplitude of 0.010 m and spatial wave length of 1.38 m. The second vertical profile is the random irregularity generated by means of the power spectral density function of a railway track as indicated by Claus and Schiehlen ([8]). This random irregularity profile has a minimum spatial wave length of 1 m and a maximum one of 80 m, in accordance with European standard (CEN EN 13848 series). The length \( L \) of the vehicle is 4 m.

Regarding the sensor models, in both cases a low frequency sinusoidal accelerometer bias has been supposed, the period being of 100 s with an amplitude value of 5% the maximum acceleration undergone by the vehicle in each case. The gyroscope bias are not estimated in these simulations, so its values were supposed to be corrupted by Guassian noise only.

Additionally, the simulations with both irregularities profile have been carried out with two different velocity functions, a constant velocity test and a sinusoidal time varying one. The variable velocity has a sinusoidal variation around the medium value between a minimum and maximum velocities of 20 m/s and 50 m/s respectively, with a period of 20 s. On the other hand, the constant velocity tests were performed at 35 m/s².

For clarity reasons, in what follows, the simulation figures present the results in two different colors; red for the Kalman filter estimates and blue for the actual parameter values.
3.1 Constant velocity tests

The vertical coordinate of the sinusoidal irregularity profile has the following expression

\[ h_k = A_h \sin(2\pi f s_k) \]  

(25)

\( s_k \) being the horizontal coordinate which is a time linear function, \( f \) the spatial frequency and \( A_h \) the amplitude. Figure 2 shows the simulation results where it is possible to see the evolution of the estimated vertical irregularity coordinate as a function of the distance travelled by the vehicle. The initial transient and steady state values are magnified in order to show the concordance in more detail.

The matching frequency content of the previous signals is depicted in Figure 3. As the horizontal coordinate \( s_k \) is a time linear function, the sinusoidal track profile has a unique spatial frequency which is exactly estimated by the filter.

Recalling Equation (7) for the acceleration model and taking into account the generated coordinate irregularity profile \( h_k = f(s_k) \), the true acceleration is determined by \( \ddot{h}_k = \frac{d^2f(s_k)}{dt^2} \). As the bias amplitude has been chosen to be a 5% of the maximum acceleration value, it must be considered that the bias values will be different for each simulation case. For the sinusoidal irregularity profile at constant velocity, the accelerometer bias estimation is depicted in Figure 4. The transient and steady state estimated values are detailed in two separated windows.

The vertical coordinate estimation of a random profile when the vehicle is advancing at constant velocity can be seen in Figure 5. Here the transient and stabilized behaviour of the filter algorithm can be appreciated in detail.

The frequency content of the random irregularity profile is richer than the corresponding to the sinusoidal case. This increases the accuracy requirements of the algorithm regarding the band of frequencies that must be estimated, where the behaviour of the gyroscope for high frequency estimation along with the accelerometer for low frequency estimation are implicitly evaluated.

Figure 6 and 7 show the matching between the actual frequency content and the bias with the estimated values for the random irregularity profile.
Figure 3. Frequency matching with the sinusoidal irregularity profile at constant velocity

Figure 4. Bias estimate at constant velocity in a sinusoidal irregularity profile

Figure 5. Constant velocity estimation of a railway random profile
Figure 6. Frequency content estimation of a the random irregularity profile at constant velocity of advance

Figure 7. Bias estimate at constant velocity in a random irregularity profile

3.2 Variable velocity tests
The last simulations in order to test the robustness of the filter algorithm has been done considering a time varying velocity function as mentioned at the beginning of this Section.

For the sinusoidal irregularity profile example, as shown in Figure 8, the vertical coordinate estimation looks similar to that of the sinusoidal irregularity profile at constant velocity previously shown in Figure 2, although the profiles are different. This fact can be easily seen by observing the frequency content of this track geometry, which is depicted in Figure 9. In this case, as the velocity of advance is not linear but follows a sinusoidal variation, the horizontal coordinate $s_k$ of Equation (25) is not a time linear function, generating an irregularity profile with a more complex spatial frequency spectrum than the previous one. The results regarding the bias estimation are shown in Figure 10. For the random profile case, the irregularity profile, frequency and bias estimation
when the vehicle is travelling at a variable velocity of advance are shown in Figure 11, 12 and 13 respectively.

**Figure 8.** Estimation of a sinusoidal irregularity for a time varying velocity of advance

**Figure 9.** Frequency content estimation of a sinusoidal irregularity profile at variable velocity of advance
Figure 10. Bias estimate for a sinusoidal irregularity profile with variable advance velocity

Figure 11. Estimation of a random irregularity track at a variable velocity of advance
Figure 12. Frequency content estimation of a random irregularity profile at a variable velocity of advance.

Figure 13. Bias estimate for a random irregularity profile with variable advance velocity.

4 CONCLUSIONS

The Kalman filter algorithm proposed in this article has been developed to fuse the signals coming from an IMU located in a vehicle that moves over two-dimensional smooth and random track profiles in order to estimate its vertical coordinates and the sensor bias. In order to test the robustness of the algorithm the most general case with a variable velocity of advance has been considered, showing a very good accordance between the estimate and actual results with both a deterministic sinusoidal track profile and a randomly generated one.

Although all the considerations have been based on two-dimensional kinematic approaches, the linearity conditions suppose that some considerations can be easily extended to the tree dimen-
Based on the good results obtained with the presented estimation algorithm, a future step in this research would be to extend its application to the dynamic case.

**References**


A Comparison between Finite Element Models and MBS Models in Automotive Safety Applications

Jörg Fehr, Christian Kleinbach

Institute of Engineering and Computational Mechanics
University of Stuttgart
Pfaffenwaldring 9, 70569 Stuttgart, Germany
[joerg.fehr, christian.kleinbach]@itm.uni-stuttgart.de

ABSTRACT

In the development of automotive safety systems finite element (FE) and multibody simulations (MBS) are used. Validated models of standard crash test dummies are available in both worlds. Here a generic side impact setup for a comparison of various models of the human body for automotive safety applications is presented. A side impact scenario is chosen, because side impacts remain very critical even in modern cars due to the narrow space available between driver and interior. The setup allows a controlled deceleration based on injury criteria defined by governmental and consumer rating institutions. First, optimal trajectories for the deceleration of a second generation European Side Impact dummy (ES-2) are developed to compare the FE and MBS models of a dummy. Afterwards, controllers are implemented which apply forces to the body regions depending on their specific injury criteria. Finally, the energy distribution across different parts of the body is presented. This allows further comparisons not only limited to dummy models but for example with modern human body models. Furthermore, an analysis of the optimal deceleration strategy is proposed as an preliminary step in the development of automotive safety systems to facilitate the design process.

1 INTRODUCTION

For obvious reasons the field of automotive safety is a well regulated environment. Regulations regarding the safety of vehicles are defined by governmental and consumer rating institutions. These regulations cover a number of precisely defined crash setups in which anthropomorphic test devices (ATD), commonly known as crash test dummies, are used to determine the effects on the human body. These ATDs are mechanical models of the human body, which are designed for specific impact types, e.g. frontal, side or rear impact. Measurement devices included in the ATDs evaluate certain forces, accelerations and deflections that were found to have a statistical relevance with respect to injuries of the human body. The development of thoracic injury and protection criteria for side impacts is explained e.g. in [1]. The limits for these injury criteria are defined by the aforementioned regulations, see also Table 1.

The development of automotive safety systems is governed by these regulations. In an iterative process designs are developed, tested in simulations or real crash tests and if the injury values breach the limits the design is improved and tested again. However, the definition of the ATD in combination with the limits for the injury values already implies an optimal trajectory for the deceleration of the dummy. In a first step, optimal trajectories for the injury values are derived. Then a controller based approach is used in the simulations to calculate the applied forces necessary to follow these trajectories. As a result optimal trajectories for applied forces, velocity and position of the ATD are available. These optimal trajectories can be used for the design of safety systems instead of the iterative approach. In addition, the trajectories represent a measure to compare different models of the human body, e.g. finite element versus multibody models or dummy versus human body models.
In the first section the ES-2 dummy and a generic side impact setup is presented, which allows the comparison of a finite element dummy model and a multibody dummy model. Then optimal trajectories for the deceleration of the ES-2 dummy in a side impact scenario are developed. Finally results are shown and discussed. A good overview regarding optimal protection with respect to injury biomechanics is given in [2]. The limiting performance, mostly in frontal impacts, based on force, deflection, integral and combined criteria for the human body is explained. In this paper the limiting performance in a side impact for a state of the art dummy model is derived, with the aim of comparing available models.

2 TEST SETUP

To be able to study the deceleration of side impact dummies across various simulation tools, a generic side impact setup was designed which can easily be set up in a finite element or multibody code. The setup is inspired by [3] and shown in Figure 1. It consists of a Heidelberg type seat, on which the dummy is sitting, respectively moving laterally with its initial velocity.

An ES-2 dummy, shown in Figure 2, is an ATD developed for side impacts and is the standard dummy used worldwide in almost all side impact crash test. With its weight of 72 kg and a sitting height of 909 mm it represents a 50th percentile male adult. The lower arms are not included to avoid contact with for example the steering wheel; this improves the repeatability of crash test results. The dummy is constructed with a metal skeleton, different foam and rubber parts are used to model the body parts of the dummy. Multiple sensors are built into the ATD to measure the influences of the crash test onto the model of the human body. The most important body part regarding injuries in crashes is the thorax. Therefore it is the body part modeled with most detail in the ES-2. The thorax of the dummy consists of three independent ribs, which are connected to the central spine box through a combination of springs and dampers. Aside from deformations of the ribs, a guiding system allows only lateral movement of the ribs.

The regulations that will be obeyed are the 2014 European New Car Assessment Program (NCAP) Criteria and the goal is to receive the highest, well known 5-star rating. This means not breaching the protection criteria listed in Table 1. To rate traumatic injuries, the Abbreviated Injury Scale (AIS) is used. The limits given in Table 1 are based on the probability of the occurrence of serious and severe injuries (AIS 3+). Complying with the higher performance criteria means the risk of serious or severe injuries is expected to be very low, which then results in a high NCAP rating. For example, when comparing the higher and lower performance limits for the thorax, the risk of serious injuries caused by compression is reduced from 17.8% to 3.3% according to the US NCAP risk curves [4].

To avoid head injuries, the head accelerations need to stay below a certain level, therefore an accelerometer is mounted in the head’s center of gravity. The head injury criterion (HIC) is an integral criterion defined by

\[
HIC_{36} = \max_{t_1, t_2: t_2 - t_1 \leq 36\text{ms}} \left\{ \left[ \frac{1}{t_2 - t_1} \int_{t_1}^{t_2} a(t) dt \right]^{2.5} (t_2 - t_1) \right\},
\]

it takes into account the severity and the duration of the impact. The second criterion \(a_{3\text{ms}}\) is simply the average acceleration over 3 ms. For the thorax, it was found that on the one hand, injuries depend on the deflection of the ribs, like e.g. fractures. On the other hand, injuries to the internal organs occur when an impact happens at a high speed and the thorax is already compressed [5]. Therefore, besides the simple rib deflection criterion (RDC), the viscous criterion (VC) is defined as

\[
VC = \frac{RDC(t)}{140 \text{ mm}} \cdot \frac{dRDC(t)}{dt},
\]

with 140 mm being half the width of the thorax of the ES-2. The deflection of the ribs is measured by linear potentiometers which are mounted at the side of the dummy opposing the impact.
The abdomen of the ES-2 dummy consists of several foam parts. Three force sensors measure the lateral forces transmitted onto the lumbar spine. The maximum of the sum of the three forces is the Abdominal Peak Force \((APF)\), which is limited by the Euro NCAP Criteria. The injury criterion for the pelvis is the pubic symphysis force \((PSPF)\). In the ES-2, the pelvic bone is modeled with a left and a right half, which are connected with a load cell at the pubic symphysis. Most of the lateral impact forces applied to the pelvic region can be measured via this sensor.

To decelerate the ATD in an optimal way, five plates are used to apply forces separately to the different body regions; the knee plate though is coupled rigidly with the pelvis plate. In Figure 1, the setup and the method of force application is depicted. This allows for an independent control of the forces applied to each body region depending on the injury criteria. It was decided not to apply force directly to the head for two reasons. First, safety systems for head protection, such as windows or curtain airbags, are deployed and designed separately from side airbags, and second, these safety systems are designed to protect the head from all possible impacts, but not to apply additional forces, as this implies high risks for head and neck injuries.

In the 2014 Euro NCAP regulations, no injury criteria for the shoulder of the ES-2 dummy are defined. This is mostly due to the design of the shoulder, which features a clavicle which rotates to the front when loaded with lateral forces. As a result of this, the sensitivity to lateral forces measured by the sensor between arm and clavicle decreases significantly. Since a criterion is needed in our setup, it was decided to use the shoulder force criterion for the new Worldwide harmonized Side Impact Dummy (WorldSID) defined in the 2015 Euro NCAP regulations. There a maximum force of \(F_{\text{shoulder,max}} = 3\) kN is defined. Similar limits where found in experiments and simulations with finite element human body models in side impact scenarios, see [6] and [7].
3 OPTIMAL DECELERATION TRAJECTORIES

The dynamic behaviour of the body parts of the ES-2 dummy and the respective deflection, velocity and force criteria form the basis for the optimal deceleration trajectories.

For the determination of the dynamic behaviour of the rib, a model identification is performed. For the model identification, the rib is extracted from the ES-2 dummy and fixed with a constraint. By examining the structure of the isolated rib as seen in Figure 3, it can be seen that the rib consists of a linear mass-spring-damper system and a rib arc with nonlinear behaviour. Therefore the response characteristic of the isolated rib is similar to a PT-2 transfer function.

![Figure 3: A rib of an ES-2 dummy.](image)

Thus a state space model of second order with the corresponding differential equation

\[ m\ddot{q} + d\dot{q} + cq = F \]  

is chosen to approximate the dynamic behaviour of the rib, where \( m \) is the mass, \( d \) the damping coefficient, \( c \) the stiffness coefficient, \( F \) is the input force and \( q \) denotes the rib deflection. To determine the model parameters the step response of the rib is simulated and a parameter identification is performed. The step response of the rib and of the identified model can be seen in Figure 4.

![Figure 4: Step response of an ES-2 rib.](image)

To determine the force to be applied by the plate, a controller based on the identified model is developed. To control the rib, a closed loop control in combination with an open loop control is implemented, see Figure 5. A PI state space controller is used, for closed loop control of the rib. For the PI state space controller, the model of the rib must be transformed into the state space model

\[ \dot{x} = A \cdot x + b \cdot u, \]

where \( A \) is the system matrix, \( b \) is the input vector, \( u \) is the control and

\[ x = \begin{pmatrix} q \\ \dot{q} \end{pmatrix} \]

is the state vector. The state feedback

\[ u_{\text{closed}} = -k^* \cdot x, \]

with the feedback vector \( k^* \) stabilizes the system while the corresponding PI controller ensures that the steady state error due to disturbances is zero. The corresponding open loop control can be written as a combination of the inverse dynamics and the state feedback:

\[ u_{\text{open}} = m\dddot{q} + (d + k_1^*)\ddot{q} + (k + k_1^*)q. \]
The control parameters can be calculated by deciding on a characteristic polynomial for the closed loop control, see [8].

To ensure that the deceleration of the dummy is as swift as possible without exceeding the NCAP criteria, an appropriate trajectory for the rib deflection needs to be determined. This trajectory $q_t$ serves as a reference signal for the rib controller and as the desired rib deflection. For an optimal deceleration the trajectory should approach the maximum steady value as fast as possible, so that the maximum stationary force can be used.

Due to the model uncertainties a 5% safety factor is used to ensure that the NCAP criteria are met. The corresponding boundary conditions for the trajectory are the rib deflection criterion

$$q_{t,\text{max}} = 22 \text{ mm} \cdot 0.95$$

(8)

and the viscous criterion with

$$q_{t,\text{max}} = \frac{140 \text{ mm}}{q_t} \cdot \frac{0.32 \text{ mm}}{\text{ms}} \cdot 0.95.$$  

(9)

To ensure a smooth trajectory it is necessary to put a boundary on the maximum force or the maximum rib acceleration. Therefore a maximum rib acceleration of

$$q_{t,\text{max}} = 60 \text{ g} \cdot 0.95 = 59 \text{ mm} \left(\frac{\text{ms}}{\text{ms}}\right)^2 \cdot 0.95$$

(10)

is imposed, in accordance with the thorax injury criteria for frontal crash. Due to the low damping and the unilateral force input it is important to ensure that the controller can follow the resulting trajectory. By neglecting the damping this leads to the conservation of energy with

$$E = \frac{1}{2} m \dot{q}^2 + \frac{1}{2} c q^2 \quad \text{and} \quad E \leq \frac{1}{2} c q_{t,\text{max}}^2.$$ 

(11)

With these constraints, the optimal trajectory $q_t$ can be computed. The constraints and the optimal trajectory can be represented and understood best in the phase portrait as shown in Figure 6. As seen in Figure 7 the controller is able to follow the trajectory without exceeding the NCAP criteria. In the first part it follows the trajectory defined by the limit for the rib acceleration, then it follows the limit curve defined by the potential energy to the maximum rib deflection, where it remains until the dummy is decelerated completely. Therefore, the trajectory never gets close to the limit of the viscous criterion.

The other body parts, shoulder, abdomen and pelvis have force based NCAP criteria, which yield more simple trajectories. The optimal deceleration of the abdomen is shown as an example in

![Figure 5: structure of the controller used for the rib.](image)
Figure 6: Boundaries for the rib trajectory.

Figure 7: Rib deflection following the trajectory.

Figure 8. The trajectory of the abdomen only needs to ensure that the maximum value of the abdominal force is lower than the limit of the NCAP criterion of 1 kN. Therefore, an exponential signal

\[ u(t) = A_{\text{max}} \cdot (1 - \exp(-t/T)) \]  

with

\[ A_{\text{max}} = 1 \text{ kN} \cdot f_{\text{safety}} \]  

is used. The time constant \( T \) is chosen such that the rise time of the trajectory is in the same order of magnitude as the optimal trajectory for the rib deflection. This ensures that all body parts are decelerated equally fast and avoids tilting of the dummy. The trajectories for the shoulder and the pelvis are chosen with the same time constant but amplitudes corresponding to their NCAP criterion. For the control a model based PID controller is used. The controller is chosen such that the overshoot is minimal and the NCAP criteria are satisfied. The safety factor \( f_{\text{safety}} \) needs to be determined depending on the body region, the model and the quality of the controller that could be achieved.

3.1 Implementation Details

For the implementation of the controller in LS-DYNA, two approaches exist: the user loading method and the curve functions. For the implementation of the user loading method, a Fortran function needs to be written and compiled. Since LS-DYNA v8.0 a direct implementation of the PID controller is available, using PIDCTL in the *DEFINE_CURVE_FUNCTION Keyword. More complex controllers can be created through combinations of standard functions in the input file. In MADYMO a PID controller is also available in the software itself. However co-simulations with MATLAB/SIMULINK are possible, where all type of controllers can be implemented easily in Simulink.

For the implementation of the controller, the current injury values are needed as an input to calculate the forces applied to the plates. Sensors need to be defined that are capable of measuring the injury values during the runtime of the simulation. In general it is easier if the measurements are based on nodal positions or coordinate systems, respectively. In the LS-DYNA model of the ES-2 used here, the abdomen forces were acquired from three contact definitions which proved difficult to access during runtime, but was finally solved by using the contact force as a nodal displacement of a mock node.
4 RESULTS

By combining the aforementioned controllers for all body parts, the dummy can be decelerated in an optimal way. However, certain characteristics pose a problem. The pelvis acts like a high-pass filter and therefore relatively high forces can be applied on the pelvis without exceeding the NCAP criteria as seen in Figure 9. Besides being unrealistic as such, these high forces can lead to an undesirable tilting of the dummy. To prevent this an upper limit of 15 kN is imposed on the pelvis force. The rotation of the shoulder, described in Section 2, is another problem. In the multibody model it is impossible to implement a closed loop control for the shoulder, because the shoulder turns to the front completely and the forces measured by the force sensor decrease uncontrollable. Instead an open loop control was used in this model.

With this modifications the optimal deceleration of the ES-2 dummy is accomplished without exceeding the NCAP criteria and without tilting. In Figure 10 and 11 the final results are displayed. The plots show the velocities of each body part in the sagittal plane for the FE and the MBS model. Starting at an initial velocity of 12 m/s, the average velocity reaches zero after 54 ms and 59 ms respectively. For the FE model the velocities of all body parts remain in a small corridor until 30 ms, for the MBS model the corridor is wider but the velocities don’t start to divert. Towards the end, the lower body parts are decelerated faster in both models due to the high pelvic forces applied.

The resulting optimal energy distribution is shown in Figure 12. The aggregated energy absorption for each body part is plotted for the FE and MBS model. It can be seen that, even though the force on the pelvis is reduced, the biggest portion of the energy is transmitted through the pelvis. Most of the energy in the upper body is transmitted through the thorax. The lowest energy portions are applied to the shoulder and abdomen region. A comparison with the overall kinetic energy shows, that most of the energy is used to decelerate the dummy. Some energy is dissipated by damping and deformation or rotational motion. In Figure 13 the energy absorption over time for each body region is depicted for the MBS model. The portion of the pelvis increases over time, while the others remain similar. The humps seen in the second half of the pelvis portion result from contact between the legs of the dummy. These humps also show up in the abdomen region.

5 CONCLUSIONS

In this paper a method is presented which, on the one hand, allows the determination of optimal deceleration forces for a complete ATD in a side impact, and on the other hand, can be used to compare different models of the human body. As an example the ES-2 Side Impact Dummy
is used. Optimal trajectories were developed for each body region and controllers were used to
determine the forces applied by the plates. Thereby the ATD is decelerated as fast as possible
without exceeding the injury criteria. These results, more specifically the force trajectories and the
energy distribution, can be used in the design of automotive safety systems. Of course this method
can be adopted for other ATDs and reduce the time for the development of new safety concepts.
This is different from the usual approach of varying parameters based on experience.
The validity of the results and the method could be proved by the implementation in two different
simulation tools. The results of the finite element model calculated with LS-DYNA differ only
slightly from the results obtained in the multibody simulations with MADYMO. Each approach
has its own advantages and disadvantages. The finite element model is a very detailed model with
more than 300 000 elements. One simulation run on a cluster using 20 CPU cores can take as long
as 8 hours. The model though has the advantage of a relative predictable response when applying
varying forces to the different body parts. This helps in the development of the control parameters.
The multibody model by contrast, consists of only 202 rigid bodies. Simulations can be run
within 5 minutes using two CPU cores of a normal workstation or notebook. The software can
be coupled to MATLAB/SIMULINK to implement the controllers fast and conveniently. But the
usage of rigid bodies makes it necessary to model deformations by allowing large penetrations of
the different bodies and plates. One result is, that the response of the model to varying force can
be very unpredictable, which makes the development of robust controllers more difficult. Also the interdependence of the body regions, as shown in Figure 13, is problematic as the controllers can’t be implemented independent from each other.

The method should be used in a very early stage to develop general safety concepts for the new challenges ahead. With autonomous driving and active safety, a large variety of crash scenarios can be imagined. Therefore, the time advantage offered by the MBS model helps to get results quickly and analyse their sensitivity to parameter changes efficiently. Especially in the evaluation of pre-crash safety concepts the time advantage is essential since simulation time is much higher and the safety concepts are still very vague at the moment. FE models allow a detailed analysis of the inside of the ATDs, as the models are very close to the reality. But due to their large computational requirements, they should be used sparsely, for example for final validations.

6 Acknowledgements
This research was partly supported by Johannes Köhler, who developed the controller structure and did most of the FE simulations during his bachelor thesis.

REFERENCES


A SIMULATION CONCEPT FOR THE AZIMUTH SYSTEM OF HORIZONTAL AXIS WIND TURBINES

Johannes Luthe*, Christoph Woernle*, János Zierath#

*Chair of Technical Dynamics
University of Rostock
Justus-von-Liebig-Weg 6, 18055 Rostock, Germany
{johannes.luthe, woernle}@uni-rostock.de,

# W2E Wind to Energy GmbH
Strandstrasse 96, 18055 Rostock
Germany
jzierath@wind-to-energy.de

ABSTRACT

Azimuth systems in wind turbines ensure the correct orientation of the nacelle relative to the current wind direction. During this yaw maneuver the systems are exposed to several dynamic influences, e.g. stochastic wind loads must be entirely compensated by the azimuth motors. Moreover, imperfections in the drive units can result in a non-uniform load distribution. Associated with this is a number of challenges that must be considered in the design phase. This contribution provides an approach to display the dynamic effects occurring in an azimuth unit during yaw maneuvers by means of multibody dynamics. Therefore, a parametric simulation model of the azimuth unit is built up as an extension to an already existing flexible multibody model of an entire 2.05 MW wind turbine. Extensive measurements are used as validation basis. Furthermore, two load cases are analysed which represent common effects occurring in practice.

1 INTRODUCTION

The global challenge of limited fossil resources causes a growing significance of renewable energies. Due to their comparatively high energy efficiency, wind turbines play an important role to accomplish the switch to sustainable energy sources. A robust design of these systems in terms of fatigue behaviour is a prior target in the dimensioning process. In general, this requires experimental analyses which in practice are not feasible over the entire product life cycle. Therefore, reliable simulation models have become indispensable for both analysing the system dynamics in the design phase on the one hand and identifying specific causes of failure in existing constructions on the other hand. At the same time constantly rising computer capacity allows an increasing depth in detail for simulations.

Figure 1: Prototype of the 2.05 MW wind turbine erected in Tarnow, Mecklenburg-Western Pomerania, Germany; a) Physical prototype; b) Mounting position of azimuth system
This contribution provides an approach to display the dynamic effects occurring in the azimuth unit, also called yaw system, of a 2.05 MW wind turbine designed by W2E Wind to Energy (W2E), see Figure 1. Therefore, multibody dynamics methods are applied which are described in detail in [5].

2 WIND TURBINE MULTIBODY MODEL

In the context of this contribution a multibody simulation model of the azimuth unit is built up as an extension to an already existing flexible model of the entire 2.05 MW turbine [3]. The prototype of this horizontal three-bladed turbine has been erected in Tarnow, Mecklenburg-Western Pomerania, Germany. It is equipped with a tubular steel tower which allows a nominal hub height of 100 m and a rotor diameter of 93 m. An innovative drive train concept is used to decouple the gearbox from undesirable rotor vibrations. Thereby a special moment bearing comprising elastic bushings prevents bending moments and shear forces to be transferred on the drive train expanding its durability.

2.1 Multibody model

For validation purposes multiple numeric models of the prototype wind turbine have been built up using different multibody codes, such as alaska/wind 8.3, SIMPACK 9.3 and MSC.Adams 2012 and Flex5, see [6]. The focus of this contribution lies on the general-purpose simulation environment MSC.Adams. A user-defined MATLAB script allows a parametric model generation. This code compiles an Adams command file in ASCII format which can be imported into the Adams/View environment subsequently. Besides a flexible tower based on finite element formulation, the multibody model comprises blades built up of discrete EULER-BERNOULLI beam models including lumped masses. This allows to reproduce nonlinear effects like centrifugal stiffening. Furthermore, a detailed drive train model is implemented, see Figure 2. To simulate the wind turbine performance under operating conditions, a discrete interface was developed which allows the incorporation of the same controller software into the simulation environment as implemented on the physical turbine. The overall wind turbine model has approximately 600 degrees of freedom.

![Figure 2: MSC.Adams model of the 2.05 MW wind turbine](image)

2.2 Aerodynamics

In order to simulate aerodynamic loads acting on the wind turbine, the model interacts with the aerodynamic code AeroDyn, developed by the National Renewable Energy Laboratories (NREL). Based on the blade element momentum theory, this code computes the aerodynamic forces and moments on the blades resulting from a user-defined wind field which is be defined by the software package TurbSim (NREL). The resulting characteristics can be influenced by a set of parameters, e.g. wind seeds, turbulence intensity and mean wind speed.
2.3 Controller systems

The controller systems of wind turbines can be classified into primary and secondary control. Primary systems ensure an efficient control of the main drive train. Extracting maximum power out of the wind and protecting electrical as well as mechanical components are two major objectives. Currently, most wind turbines are operated with linear closed loops, using the generator speed as plant input and, dependent on the operating mode, the generator torque or the blade pitch angle as actuating signal. The plant output of the primary system, however, is subsequently used as reference signal for the frequency converters of the generator and the pitch drives, respectively. Controllers required for these peripheral devices are referred to as secondary control systems in the context of this contribution. Accordingly, the azimuth control in section 3.3 can be classified as a secondary control system.

3 AZIMUTH DRIVE SYSTEM

The azimuth unit is a compound of several subsystems whose main task is to ensure the correct orientation of the nacelle relative to the current wind direction. More information on general implementation of azimuth systems in wind turbines is provided by [1] and [2].

3.1 General structure

The analysed azimuth system of the 2.05 MW turbine consists of four electrical drive units with multistage planetary gears and integrated brake assemblies. The exact number of drive units is generally determined by the torque required for a yaw maneuver. Additionally, four hydraulic brake units keep the nacelle in position after a succeeded yaw maneuver. The drive units are installed in the nacelle and mesh with a gear ring mounted on the tower head, see Figure 3. Ensuring a homogeneous load transmission, this design leads to a mechanical system which is referred to as redundantly actuated, however.

Figure 3: Azimuth drive unit; a) View inside nacelle; b) View inside tower head

Additionally, the drive units possess specific flexibilities in the gear stages due to their high transmission ratio in practice. Associated with this is a number of challenges that must be considered for yaw maneuvers. For example, not all of the drive units will usually speed up simultaneously, once the control command to yaw is given. Among other issues, this may cause interlocking between the drive units which can subsequently lead to significant peaks in the drive train loads, see Figure 4.
A basic yaw maneuver can be subdivided into different sections. At first, both motor and azimuth brakes are released. However, the azimuth brakes are only partially opened to increase damping during the yawing motion. Subsequently the azimuth motors speed up following reference speed curves which are formed like smoothed trapezoidal functions with a plateau at rated motor speed, see Figure 6b.

The aim of this contribution is to built up a simulation model which is able to reproduce different dynamic effects occurring in yaw maneuvers in order to consider possible additional loads in the design phase already.

![Figure 4: Drive train interlocking during the first section of an exemplary yaw maneuver of the 2.05 MW turbine; a) Measured speeds of two specific drive units; b) Measured torques of the same drive units](image)

3.2 Multibody model

The multibody model of the azimuth system is built up parametrically using MSC.Adams. Its topology is shown in Figure 5. In physical wind turbines, the different electrical, hydraulic and mechanical components of azimuth drives interact with each other in a complex way. However, the focus of this contribution lies on the mechanical behaviour and the resulting load transmission of the azimuth system. Therefore, the influence of electrical and hydraulic components is not considered. The motor units are modelled as rigid bodies with respective inertia properties available by the manufacturers. Effects like bearing friction and electrical losses are neglected within the investigations described in this contribution.

![Figure 5: Mounting position and multibody topology of the azimuth system](image)
Assuming equivalence of the kinetic energy at the respective in- and output, the downstream planetary gears are reduced to a single gear stage. Thus, frictional losses of the gear pair contacts are regarded to have minor effects only. This allows to model the entire planetary gear masses as a single rigid body, referring the inertia properties to the output. Specific flexibilities, which are present in the gear stages due to elastic deformations of the gear pairs, can be reproduced by a torsional spring. Furthermore, imperfections in physical planetary gears can result in internal clearance. Both drive train flexibility and internal clearance are represented by nonlinear spring characteristics recorded by W2E. The assumptions made lead to the approach of reproducing the dynamic behaviour of the azimuth drive units by kinematically coupled mass-spring oscillators, see Figure 5.

The influence of the brake units is represented by external forces, which are computed under the assumption of COULOMB friction. Appropriate parameters are available from the manufacturers. After all, the entire simulation model is fully parametrised using MATLAB. Thus, the possibility is given to adapt the azimuth system to different wind turbine types.

3.3 Controller

Due to its four kinematically coupled drive units, the azimuth system of the 2.05 MW turbine is a redundantly actuated system, i.e. it can be internally loaded without generating a motion. In general, a superior controller level can help to minimise the internal loading. However, the azimuth drive units are controlled individually in practice. Variations in current and voltage supply or clearance in the gear stages can therefore result in interlocking during a yaw maneuver.

The drive units of the physical turbine, comprising frequency converters and asynchronous motors, possess speed controllers with underlaid current and voltage control loops. However, W2E has no information about the exact structure and parametrisation of this secondary control system.

In order to perform simulation studies, a suitable controller model is needed which comprises comparable properties to that one operating the physical azimuth drives. Accordingly, the objective in this contribution is not to find the optimal controller design for the azimuth system. More important is the ability of reproducing dynamic behaviour of the physical azimuth control system. A crucial point for the controller design is the assumption of synchronous motions of the four kinematically coupled drive units. This refers to the physical controllers where each motor is individually controlled by feedback of the corresponding motor speed only. There is no coupling of speed feedback between the motors nor the angular speed of the nacelle is fed back. This allows to model each of the drive units individually as a two-mass oscillator, see Figure 6a. Effects like internal clearance and friction are not considered resulting in a linear system plant definition.

Figure 6: a) Reduced azimuth drive unit model for the controller design; b) Closed control loop for an individual azimuth motor
All inertia, stiffness and damping parameters are reduced to the motor shaft level. The moments of inertia $J'_1$ and $J'_2$ can be calculated as

$$J'_1 = J_M, \quad J'_2 = \frac{J_p}{i_p} + \frac{J_N}{4(i_G i_p)},$$  \hspace{1cm} (1)$$

where $J_M, J_p$ and $J_N$ are the moments of inertia of the motor shaft, the planetary gear and the nacelle, respectively. These are referred to the motor shaft level considering the specific transmission ratios of the planetary gear stages $i_p$ and of the gear ring $i_G$. The variable $M'_1$ represents the respective motor torque, whereas $M'_2$ merges external wind loads acting on the nacelle and friction influences of the azimuth brakes. Thus, $M'_2$ is treated as disturbance variable.

Setting up the equations of motion for this system and defining the applied torques $M'_1$ and $M'_2$ as inputs and the resulting angular speeds $\omega_1 = \dot{\phi}_r$ and $\omega_2 = \dot{\phi}_r$ as outputs leads to the transfer function matrix $G(s)$ in Laplace domain

$$[\omega_1(s) \omega_2(s)] = \frac{1}{N} \begin{bmatrix} J'_2 s^2 + d' s + c' \\ d' s + c' \\ J'_1 s^2 + d' s + c' \end{bmatrix} \begin{bmatrix} M'_1(s) \\ M'_2(s) \end{bmatrix}, \text{ with } N = J'_1 J'_2 s^3 + (J'_1 + J'_2)(d' s^2 + c' s).$$  \hspace{1cm} (2)$$

Assuming that sensors and actuators are collocated at the motor shaft, the resulting control loop can be transferred to a SISO control loop, see Figure 6b. The plant transfer function from $M'_1(s)$ to $\omega_1(s)$ is given by the upper left entry of $G(s)$, henceforth simply referred to as $G(s)$,

$$\omega_1(s) = G(s) M'_1(s), \quad \text{where } G(s) = \frac{1}{N} \left( J'_2 s^3 + d' s^2 + c' \right).$$  \hspace{1cm} (3)$$

The speed controller is designed as PI controller, since this type is commonly implemented in asynchronous azimuth motors. Its transfer function in Laplace domain is given by

$$G_{PI}(s) = k \left( 1 + \frac{T_I s}{T_{Is}} \right).$$  \hspace{1cm} (4)$$

Delaying characteristics of the frequency converters can be approximated by first-order delay elements (PT1),

$$G_{FC}(s) = \frac{1}{T_{FC}s + 1},$$  \hspace{1cm} (5)$$

with a time constant of $T_{FC} = 0.001 \text{s}$. The entire closed loop transfer function then adds up to

$$G_W(s) = \frac{G_{PI}(s) G_{PT1}(s) G(s)}{1 + G_{PI}(s) G_{PT1}(s) G(s)}. \hspace{1cm} (6)$$

During yaw maneuvers the azimuth drive controllers are exposed to disturbance influences e.g. by wind loads. To compensate steady-state deviations, the parameters $k$ and $T_I$ are defined according to the symmetrical optimum,

$$T_I = 8 T_{FC}, \quad k = \frac{T_M}{4 k_M T_{FC}},$$  \hspace{1cm} (7)$$

where $k_M$ represents the gain and $T_I$ the integral time constant of the plant function (3). The symmetrical optimum is a commonly applied procedure for optimising controllers in variable-speed asynchronous motor, such as azimuth motors in physical wind turbines, see [4].
4 VALIDATION

During every yaw maneuver of the 2.05 MW wind turbine prototype, the actual torques and speeds of each azimuth drive unit are recorded by W2E. In addition, the mean wind speed is registered in a temporal resolution of 10 minutes. These data build up a profound basis to validate the multibody model of the azimuth drive unit.

The flexible model of the entire 2.05 MW turbine has already been validated separately, see [6]. In a next step, the azimuth drive is integrated as a subsystem to simulate yaw maneuvers. The simulations are performed with due regard to wind loads on the turbine about its vertical axis that have to be compensated by the azimuth system. However, the actual effective wind field in front of the physical turbine cannot be reproduced exactly in simulations because of missing information about its three-dimensional properties. Therefore, the software packages TurbSim and AeroDyn are applied to generate user defined stochastic wind fields. For the validation process the mean wind speeds of the simulated wind fields are set identical with the measured data. Simulation studies have been performed at three mean wind speeds, 7.4 m/s, 8.5 m/s and 9.9 m/s. However, this contribution is focused on yaw maneuvers at 9.9 m/s wind speed which results in the highest loads on the azimuth system. Furthermore, a turbulence intensity of 8 % is assumed to reproduce the real wind regime at hub height approximately.

Analysing a number of measured data sets proves that the rotations of the four azimuth motors are not entirely synchronous during a yaw maneuver. This results in different motor torque levels and thus in interlocking of the drive trains, see Figure 4. For a conservative load assumption the motor producing the highest torque is only regarded henceforth. For the validation all drive units are operated synchronously during the simulation resulting in identical motor speeds and torques. The measured and simulated azimuth motor speeds and torques for an exemplary yaw maneuver are shown in Figure 7.

![Simulation and Measurements](a) Speed in [rpm] vs. Time in [s]
![Simulation and Measurements](b) Torque in [Nm] vs. Time in [s]
![Simulation and Measurements](c) Speed in [rpm] vs. Time in [s]
![Simulation and Measurements](d) Torque in [Nm] vs. Time in [s]

Figure 7: Yaw maneuver at a mean wind speed of 9.9 m/s; a-b) Measured and simulated motor speeds for the azimuth drive producing the highest motor torque; c-d) Measured and simulated motor speeds and torques with decreased gain factor $k$ in the PI controller

The simulation results generally show a good qualitative agreement with the measured data, see Figures 7a and 7b. However, the measured motor speeds of the actual drives underlie higher variations than the simulated results. Reasons for this can be manifold. Generally, internal clearance in gear stages can affect the respective speeds. The multibody simulations show that this effect is
of rather small significance, however. More importantly, the resulting speed strongly depends on the choice of the controller parameters. Decreasing the gain factor $k$ in the PI controller leads to higher oscillations in the resulting speed and thus to lower torque levels, see Figures 7c and 7d. Therefore, the parameter $k$ needs to be carefully chosen in order to get reasonable results for both motor speeds and torques.

The simulation is able to reproduce the measured motor torque curves qualitatively in a good way. However, quantitatively the simulated results show an average downward deviation of about 54% compared to measurements, as shown in Figure 7b. Again, reasons can be found in different fields. In contrast to the simulation, physical azimuth drives do generally not rotate synchronously due to parasitic influences. As a consequence, this leads to different motor torque levels in the respective drive units. However, the exact amount of deviation varies stochastically and thus the recorded torques can only be partially reproduced in a simulation. Another important influence on the results is given by bearing friction and splashing losses in the planetary gears. Neglecting these effects in the simulation leads to a lower yawing resistance and thus to lower required motor torques in the simulated azimuth unit compared to the physical system. In addition, the unknown wind regime in front of the prototype turbine can be identified as further influencing factor on the results. Nevertheless, this influence is of stochastic matter and can again only be partially reproduced in a simulation.

Measurements and subsequent analysis reveal that the controllers implemented in the physical azimuth drive units do not show optimal command response under wind loads. Thus, the objective in this contribution is not to design the optimal controller for the modelled plant, but one, which is able to reproduce the dynamic properties in a satisfying way. A PI controller with reasonably adapted parameters proves to be appropriate. In summary, the multibody model of the azimuth system can be adequately validated on a basis of extensive measurements by W2E.
5 LOAD CASES

During the practical operation of azimuth systems the individual components are exposed to different interfering dynamic influences. To ensure a robust design, possible additional loads must be analysed by a number of load cases. In the context of this contribution two main effects are presented which can both be ascribed to deviations in the speeds of the azimuth drives commonly occurring in practice. To display the resulting effects, a reference simulation is done beforehand.

5.1 Reference simulation

The reference simulation of an exemplary yaw maneuver is performed at a mean wind speed of 9.9 m/s and a turbulence intensity of 8%. At first, the azimuth and motor brakes are released before the drive units speed up following a reference speed curve, see Figure 8a. During the yaw maneuver the motions of all drive units are perfectly synchronous. Unsymmetrical wind loads induce a temporally variable yaw momentum to the nacelle which must be entirely compensated by the azimuth brake units and the azimuth motors, respectively. This leads to motor torques up to 35 Nm, see Figure 8b. Figure 8c shows that the drive trains are loaded due to elastic deformation in the gear stages. Finally the nacelle is yawed by 20 deg.

Figure 8: Reference simulation; yaw maneuver at mean wind speed of 9.9 m/s and turbulence intensity of 8%; a) Actual speed of all azimuth motors; b) Resulting torque of all azimuth motors; c) Resulting torsional moment in all drive trains; d) Yaw angle of the nacelle
5.2 Delayed start of an individual motor

Due to disturbances, in practice, not all azimuth motors start to speed up simultaneously and do not run perfectly synchronously during the yawing period. Exemplarily, motors II-IV start to move the nacelle while motor I is delayed by 1s, see Figure 9a. However, due to kinematic constraints motor I is dragged as well. Thus, its internal speed controller tries to impede a movement and consequently builds up a counter-torque, see Figure 9b. This results in significant motor and drive train overloads compared to the reference load case considered in Figure 8. As shown in Figure 9c, the different load levels in the individual drive trains are not adjusted during the entire yaw maneuver. This effect can be ascribed to the decentralised azimuth controller structure. Each drive unit is equipped with a separate controller unit that does not interact with the others, respectively. As a consequence, a superior controlling level is required to reduce interlocking of the drive trains.

![Figure 9: Starting and stopping of motor I delayed by 1 s; a) Actual speeds of the azimuth motors; b) Resulting torques of the azimuth motors; c) Resulting torsional moments in the drive trains; d) Yaw angle of the nacelle](image)

5.3 Deviation of actual speeds in the drive units

A further effect commonly occurring in the practical operation of azimuth systems is the deviation of the actual speeds in the drive units. To illustrate the impact qualitatively, the reference speed of motor I is exemplarily reduced by 30 rpm, see Figure 10a. The controller of motor I tries to maintain the reduced speed during the yaw maneuver, whereas the other controllers force to speed up, see Figure 10b. This results in considerable, temporarily increasing motor loads, which are subsequently transferred to the downstream planetary gears as shown in Figure 10c. After the succeeded yaw maneuver both motor and azimuth brakes are closed again, whereby the load level in the drive trains are maintained. Subsequently, the motors are relieved by setting the applied torque to zero.

![Figure 10: Deviation of actual speeds in the drive units; a) Actual speeds of the azimuth motors; b) Resulting torques of the azimuth motors; c) Resulting torsional moments in the drive trains; d) Yaw angle of the nacelle](image)
Figure 10: Reference speed of motor I reduced by 30 rpm; a) Actual speeds of the azimuth motors; b) Resulting torques of the azimuth motors; c) Resulting torsional moments in the drive trains; d) Yaw angle of the nacelle

6 CONCLUSIONS

Azimuth systems in wind turbines are exposed to several dynamic influences during a yaw process. Stochastic wind loads, imperfections in gear stages and a decentralised controller structure can cause overloads in both the azimuth motors and the downstream planetary gears. These additional loads must be considered in terms of a robust design.

The present contribution proposes a way of displaying these characteristic effects by means of a multibody simulation. Therefore, the azimuth system of a 2.05 MW wind turbine prototype erected by W2E is built up parametrically using MATLAB and MSC.Adams. Extensive measurements conducted by W2E build up a profound basis for an adequate validation. Forward dynamics simulations of yaw maneuvers require suitable controllers in the azimuth drive units. However, wind turbine manufacturers do usually have no information on the exact structure and parametrisation of physical controllers in secondary control systems as the azimuth unit. The present contribution shows that a PI controller is sufficient to reproduce the major dynamic effects in the azimuth system. However, parameters must be carefully adapted to the respective application.

Two exemplary load cases show qualitatively, how common disturbances in the practical operation of azimuth systems can result in interlocking of the drive trains and therefore in considerable overloads. Thus, multibody simulations can be successfully used to show the behaviour of the controlled system in practice.

REFERENCES


A transcription method for optimal control problems in multibody dynamics

C. Becker, P. Betsch
Institute of Mechanics
Karlsruhe Institute of Technology (KIT)
Otto-Ammann-Platz 9, 76131, Karlsruhe, Germany
christian.becker@kit.edu, peter.betsch@kit.edu

Abstract
The present work deals with the optimal control of mechanical systems in terms of a direct transcription scheme, in this approach the formulation of the embedded discrete equations of motion has a direct impact on the structure of the necessary conditions of optimality. In this regard, we will focus on the control of a Cosserat point (pseudo-rigid body), which represents a finite-dimensional model for a deformable body. This model problem already features key structural properties of more complicated mechanical systems such as nonlinear elastodynamics and structural dynamics. In contrast to the continuum theory the equations governing the motion of a Cosserat point consist of ordinary differential equations. Due to its relative simplicity the theory of a Cosserat point is deemed to be especially well-suited to convey main ideas of the design of Energy-Momentum schemes. In addition to that, the theory of a Cosserat point paves the way to rigid body dynamics, where additional internal constraints are imposed, leading to differential-algebraic equations (DAEs) governing the motion of a rigid body. To this end the Cosserat point is a convenient mechanical object to establish both a link between rigid and flexible body dynamics as well as between ODE- and DAE-structured state equations.

Keywords: Optimal Control, Cosserat Theory, Transcription Scheme.

1 Introduction
The focus of this work is the detailed direct transcription of a set of discrete equations, based on a template scheme, governing the motion of an exemplary mechanical system. Here, the emphasis lies on the attitude maneuver of a spacecraft in a circular orbit under the influence of a Newtonian gravitational force field exerted by a spherically symmetric rigid central body, see also [5]. As the preservation of resources becomes increasingly important the subject of optimization is the required control effort to move the system from a specific initial to a specific terminal state. In this regard the satellite is modeled as a pseudo-rigid body, based on the Cosserat theory, see also [9].

2 Dynamics
The assumption of spatially homogeneous deformations is imposed by considering affine deformation maps of the form

\[ x = \Phi (X, t) = \bar{x} + F(t)(X - \bar{X}) \]  

(1)

Here, material points in the reference configuration \( B_0 \subset \mathbb{R}^3 \) are denoted by \( X \in B_0, \bar{X} \in B_0 \) is the center of mass, and \( \bar{x}(t) \in B_t \) denotes the corresponding placement in the configuration \( B_t \subset \mathbb{R}^3 \) at time \( t \in \Omega, \Omega = [0, T]\), the time domain of interest. Moreover, \( F(t) = D\Phi_t(X) \) is the deformation gradient, where \( \Phi_t(X) = \Phi(X,t) \).

To obtain equations of motion of a pseudo-rigid body in a form which belongs to the theory of a Cosserat point dealt with in [9], we write the homogeneous deformation gradient as

\[ F(t) = d_i(t) \otimes D^i \]

(2)

where \( d_i(t) \in \mathbb{R}^3 \) are three director vectors that in general rotate, stretch and shear with the body in its motion, see also Figure 1. The directors are subject to the requirement that \((d_1 \times d_2)\cdot\).
Figure 1. Illustration of the Cosserat point

\( d_3 > 0 \), which is consistent with the nonsingularity of the homogenous deformation gradient \( F \). Furthermore, we introduce the vectors \( D^i \in \mathbb{R}^3 \) constituting the director triad in the reference configuration \( B_0 \). In particular, \( D^i \) represent a basis fixed in space whose origin coincides with the center of mass. The corresponding coordinates will be denoted by

\[
X^i = D^i \cdot (X - \bar{X}) \tag{3}
\]

Without loss of generality we assume that the directors in the reference configuration are mutually orthonormal, that is \( D^i \cdot D^j = \delta^{ij} \), where \( \delta^{ij} \) is the Kronecker delta. Note that the last assumption implies \( D_i = D^i \) as well as a stress-free reference configuration of the Cosserat point. Inserting (2) into (1) and taking into account (3) yields

\[
\bar{x}(X,t) = \bar{x}(t) + X^id_i(t) \tag{4}
\]

Note that, due to the kinematic assumption (1) or (4), the free Cosserat point has \( n = 12 \) degrees of freedom (DOFs). At this point the equations of motion pertaining to pseudo-rigid bodies can be derived from the principle of virtual work for a general deformable continuum.

\[
\delta \bar{x} \cdot (M \ddot{\bar{x}} - \bar{f}) + \delta d_i \cdot (E^{ij}_0 \ddot{d}_j - f_i) = 0 \tag{5}
\]

where the force vectors \( \bar{f}, f_i \in \mathbb{R}^3 \) conjugated to the respective variation can be partitioned into

\[
\bar{f} = f^{\text{act}} \quad \quad f_i = f_i^{\text{act}} - f_i^{\text{int}} - f_i^{\text{grav}} \tag{6}
\]

with actuation forces \( f^{\text{act}}, f_i^{\text{act}} \in \mathbb{R}^3 \), internal forces \( f_i^{\text{int}} \in \mathbb{R}^3 \) and gravitational forces \( f_i^{\text{grav}} \in \mathbb{R}^3 \). Moreover \( E^{ij}_0 \) are the cartesian components of the referential Euler tensor \( E_0 \) given by

\[
E_0 = \int_{B_0} \rho_0 \left( X - \bar{X} \right) \otimes \left( X - \bar{X} \right) \, dV
\]

\[
E^{ij}_0 = D^i \cdot E_0 D^j = \int_{B_0} \rho_0 X^i X^j \, dV \tag{7}
\]

Considering the following discussion in regard to the attitude control of a satellite, which follows a prescribed path in a circular orbit, we can omit the equations of motion in regard to the position...
of the center of mass in (5), i.e. set $\delta \ddot{x} = 0$. Moreover we will only deal with the director forces in (6)$_2$ within the next sections. Accordingly, the second-order ODEs

$$E^{ij}_0 \dot{\mathbf{d}}_j = f_i$$

(8)
governing the motion and deformation of the Cosserat point relative to the center of mass will serve as state equations in the sequel.

**Internal forces.** We introduce the total strain energy

$$U = \int_{B_0} W(G) \, dV = V_0 W(G)$$

(9)

Here, we consider a St. Venant-Kirchhoff material with strain energy density

$$W(G) = \frac{1}{2} \left( \text{tr} G + \mu \text{tr} G \right)$$

(10)

By inserting (2) into the expression for the right Cauchy-Green tensor $C = F^T F$, the Green-Lagrangean strain tensor yields

$$G = \frac{1}{2} (C - I) = \gamma_{ij} D^i \otimes D^j$$

(11)

where the cartesian components of the Green-Lagrangean strain tensor are given by

$$\gamma_{ij} = \frac{1}{2} (d_{ij} - \delta_{ij}) \quad \text{with} \quad d_{ij} = \mathbf{d}_i \cdot \mathbf{d}_j$$

(12)

The second Piola-Kirchhoff stress tensor is given by

$$S = D W(G) = \lambda \left( \text{tr} G \right) I + 2\mu G$$

(13)

Since we have assumed that the director triade $D^i$ in the reference configuration is orthonormal, it suffices to consider the cartesian components of $S$. Accordingly, we obtain

$$S^{ij} = D^i \cdot D W(G) D^j = \lambda \gamma_{kk} \delta_{ij} + 2 \mu \gamma_{ij}$$

(14)

Next we consider

$$\frac{d}{dt} U(G) = DU(G) : \dot{G} = \frac{1}{2} V_0 D W(G) : \dot{d}_{ij} D^i \otimes D^j = \frac{1}{2} V_0 S^{ij} \ddot{d}_j$$

(15)

which under consideration of the symmetry of $S^{ij}$ can be recast into

$$\frac{d}{dt} U(G) = V_0 S^{ij} \mathbf{d}_j \cdot \dot{\mathbf{d}}_i = \bar{S}^{ij} \mathbf{d}_j \cdot \dot{\mathbf{d}}_i$$

(16)

Eventually the internal director forces yield

$$f_{i}^{\text{int}} = \bar{S}^{ij} \mathbf{d}_j$$

(17)

**Gravitational forces.** The pseudo-rigid body of mass $m$ is assumed to be under the influence of a Newtonian gravitational force field exerted by a spherically symmetric rigid body of mass $M$, located at the fixed position $\varphi_M$. As can be seen in Figure 2, the position vector of the pseudo-rigid body is denoted by $\varphi$. The corresponding potential of the external gravitational field is

$$U_{\text{grav}} = -\frac{GMm}{R} + \frac{GM}{2R^2} \left( \text{tr} \mathbf{E} - 3 e_r \cdot \mathbf{E} e_r \right)$$

(18)
with
\[ R = \|\varphi - \varphi_M\| \quad e_r = \frac{\varphi - \varphi_M}{\|\varphi - \varphi_M\|} \] (19)

together with the gravitational constant \( G \) and the spatial Euler tensor \( E = F E_0 F^\top \). Note that a more detailed treatment of the potential can be found in [8]. For our purpose we recast the expression (18) into

\[ U_{\text{grav}}(F) = -\frac{G M m}{R} + A : E \]
\[ = -\frac{G M m}{R} + \text{tr} \left( F^\top A F E_0 \right) \] (20)

with
\[ A = \frac{G M}{2 R^3} (I - 3 e_r \otimes e_r) \] (21)

The derivative in respect to the deformation gradient \( F \) yields

\[ DU_{\text{grav}}(F) = A F E_0 + A^\top F E_0^\top \]
\[ = 2 A F E_0 \]
\[ = 2 A \left( d_i \otimes D^i \right) E_0^{j k} \left( D^j \otimes D^k \right) \]
\[ = 2 A E_0^{ij} d_i \otimes D^j \] (22)

where the symmetry of \( E_0 \) and \( A \) has been taken into account. Similar to (15) we consider

\[ \frac{d}{dt} U_{\text{grav}}(F) = DU_{\text{grav}}(F) : \dot{F} = DU_{\text{grav}}(F) : \left( d_i \otimes D^i \right) = \dot{d}_i \cdot DU_{\text{grav}}(F) D^i \] (23)

Consequently the gravitational director forces yield

\[ f_{i}^{\text{grav}} = DU_{\text{grav}}(F) D^i = 2 A E_0^{j k} \left( d_j \otimes D^k \right) D^i = 2 A E_0^{ij} d_j \] (24)

**Actuation forces.** We introduce the following actuation forces accounting for the deformability of the pseudo-rigid body

\[ f_{i}^{\text{act}} = U^{ij} d_j \] (25)
Here, \( U^{ij} \in \mathbb{R}^{2n} \) represent \( n_u = 9 \) independent stress-type actuation inputs. Note, that in the context of optimal control theory these quantities will also be referred to as control inputs.

### 3 Optimal Control

The optimal control task at hand can be stated as follows: Find the trajectory, comprising both the state variables \( x(t) \) and control inputs \( u(t) \) over the time domain \( \Omega = [0, T] \), that minimizes the performance criteria \( L_\Omega \) and satisfies the terminal state constraints \( \Psi \) at time \( T \). At this point we introduce the augmented performance functional

\[
J = \eta \cdot \psi \bigg|_T + \int_\Omega L(u(t)) + \mu(t) \cdot g(x(t), \ddot{x}(t), u(t)) \, dt
\]

in the continuous setting. Here, the state vector comprises both the directors and director-velocities, i.e. \( x(t) = \{ d_i, v_i \} \), \( \eta \) are the Lagrangian multipliers corresponding to the terminal state constraints \( \Psi \) and \( \mu \) are the costate variables conjugated to the set of state equations \( g(x(t), \ddot{x}(t), u(t)) \). The performance criteria \( L_\Omega \) together with the performance index \( L \) is given by

\[
L_\Omega = \int_\Omega L(u) \, dt \quad \text{and} \quad L(u) = \frac{1}{2} u \cdot u
\]

and accounts for the control effort over the time domain \( \Omega \).

In line with the direct transcription approach in optimal control theory (see also [2], [3], [4], [6], [7]), the discretization of the augmented performance functional in (26) yields

\[
J^h = \eta \cdot \psi \bigg|_T + \sum_{n=1}^N \left( L^h(u_n) h + \theta_n \right) \quad \theta_n = \mu_n \cdot g^h_n(x_{n-1}, x_n, u_n)
\]

(28)

of the discrete state equations by applying an EM-scheme, see also [1], to (8)

\[
g^h_n = \begin{bmatrix}
    h^{-1} (d_{i_n} - d_{i_{n-1}}) - v_{n-\frac{1}{2}} \\
    h^{-1} E_0^{ij} (v_{j_n} - v_{j_{n-1}}) - f_{i_{n-1},n}
\end{bmatrix}
\]

(30)

Here, \( h = t_n - t_{n-1} \) denotes the length of equidistant time intervals over the time domain \( \Omega \) and \((\bullet)_{n-\frac{1}{2}}\) denotes the mean value of the quantity \((\bullet)\), that is

\[
(\bullet)_{n-\frac{1}{2}} = \frac{1}{2} ((\bullet)_{n-1} + (\bullet)_{n})
\]

(31)

Analogously to (6)_2, we partition the discrete director forces as follows

\[
f_{i_{n-1},n} = f^{\text{act}}_{i_{n-1},n} - f^{\text{int}}_{i_{n-1},n} - f^{\text{grav}}_{i_{n-1},n}
\]

(32)

In favour of a more detailed depiction of the conditions of optimality we partition the costate variables, i.e. \( \mu_n = \{ \mu^q_{i_n}, \mu^v_{i_n} \} \). Now, (28)_2 together with the discrete state equations (30) can be written out as

\[
\theta_n = \mu^q_{i_n} \cdot \left( h^{-1} (d_{i_n} - d_{i_{n-1}}) - v_{n-\frac{1}{2}} \right) + \mu^v_{i_n} \cdot \left( h^{-1} E_0^{ij} (v_{j_n} - v_{j_{n-1}}) - f_{i_{n-1},n} \right)
\]

(33)

Eventually the variation of (28) together with (33) yields the discrete necessary conditions of optimality (DNOC), where in favour of notation the arguments for the partitioned expressions are omitted at this point and in the sequel

\[
\delta J^h(\Phi; \delta \Phi) = \delta J^{\text{state}} + \delta J^{\text{costate}} + \delta J^{\text{control}} + \delta J^{\text{terminal}}
\]

(34)
with
\[
\delta J^{\text{state}} = \sum_{n=1}^{N} \delta \mu_{n} \cdot g_{n}
\]
\[
\delta J^{\text{costate}} = \sum_{n=1}^{N} \delta d_{i_{n}} \cdot \left[ h^{-1} \left( \mu_{q_{n+1}}^{q} - \mu_{q_{n}}^{q} \right) + \left( \nabla d_{i_{n}} f_{j_{n-1,n}} \right)^{T} \mu_{j_{n}}^{q} + \left( \nabla d_{i_{n}} f_{j_{n,n+1}} \right)^{T} \mu_{j_{n+1}}^{q} \right] + \sum_{n=1}^{N} \delta v_{i_{n}} \cdot \left[ h^{-1} E_{ij}^{n} \left( \mu_{j_{n+1}} - \mu_{j_{n}} \right) + \mu_{i_{n+1}}^{q} \right]
\]
\[
\delta J^{\text{control}} = \sum_{n=1}^{N} \delta u_{n} \left[ \nabla u_{n} \mathcal{L}^{h} h - \left( \nabla u_{n} f_{j_{n-1,n}} \right)^{T} \mu_{j_{n}}^{q} \right]
\]
\[
\delta J^{\text{terminal}} = \delta x_{N} \cdot \left( \nabla x_{N} \Phi \right)^{T} \eta + \delta \eta \cdot \Phi
\]  
(35)

Here, in sequential order (35)\textsubscript{1} are the discrete state equations, (35)\textsubscript{2} are the discrete costate equations, (35)\textsubscript{3} are the discrete control equations and (35)\textsubscript{4} depict the terminal state constraints together with appended costate expressions. Moreover, the vector
\[
\Phi = \left\{ \mu_{n}, d_{i_{n}}, v_{i_{n}}, u_{n}, \eta \right\} \quad \forall \quad t_{n} \in \Omega
\]  
(36)

condenses all unknowns over the time domain for the optimal control problem at hand. Note, that the control inputs \( u_{n} \) and costate variables \( \mu_{n} \) correlate rather to the respective time intervall than a discrete time node \( t_{n} \). Next, we will elaborate on the discrete director forces and their respective impact on the DNOC in a similar way to section 2.

**Internal forces.** The contribution of the internal director forces to the DNCO yield
\[
\delta J_{\text{int}}(\Phi; \delta \Phi) = \delta J_{\text{int}}^{\text{state}}(\Phi; \delta \Phi) + \delta J_{\text{int}}^{\text{costate}}(\Phi; \delta \Phi)
\]  
(37)

where we distinguish between contributions to the state and costate equations in (35)\textsubscript{1} and (35)\textsubscript{2} respectively. In view of the contribution to the state equations we obtain
\[
\delta J_{\text{int}}^{\text{state}}(\Phi; \delta \Phi) = \sum_{n=1}^{N} \delta \mu_{i_{n}}^{q} \cdot \left[ - f_{i_{n-1,n}}^{\text{int}} \right]
\]  
(38)

The discrete variant of the internal forces in (17) is given by
\[
f_{i_{n-1,n}}^{\text{int}} = \tilde{S}_{h_{n-1,n}}^{ij} d_{j_{n-\frac{1}{2}}}
\]  
(39)

with
\[
\tilde{S}_{h_{n-1,n}}^{ij} = V_{0} S_{h_{n-1,n}}^{ij}
\]  
(40)

Note, that the subscript index (\(\bullet\))\textsubscript{A} indicates the modification of the mid-point rule in regard to the applied EM-scheme according to the method developed by Simo and Tarnow (1992), see [10]. In particular, the discrete components of the second Piola-Kirchhoff stress tensor are
\[
S_{h_{n-1,n}}^{ij} = \lambda_{g} k_{n-\frac{1}{2}} \delta_{ij} + 2 \mu_{r} g_{ij, n-\frac{1}{2}}
\]  
(41)

where the cartesian components of the Green-Lagrangian strain tensor assume the form
\[
\gamma_{ij, n-\frac{1}{2}} = \frac{1}{2} \left( d_{ij, n-\frac{1}{2}} - \delta_{ij} \right)
\]  
(42)
together with the average of the metric coefficients
\[
d_{ij,n-\frac{1}{2}} = \frac{1}{2} \left( d_{ij,n-1} + d_{ij,n} \right)
\]
\[
= \frac{1}{2} \left( d_{i,n-1} \cdot d_{i,n-1} + d_{i,n} \cdot d_{in} \right)
\]
(43)

Note that this is in contrast to the mid-point rule, in which
\[
d_{MP,ij,n-\frac{1}{2}} = d_{i,n-\frac{1}{2}} \cdot d_{j,n-\frac{1}{2}}
\]
(44)

would have to be applied. At this point and in the sequel the notation \((\bullet)^{MP}\) defines the standard mid-point evaluation. In regard to the contribution to the costate equations (35)
\[
\delta J_{\text{costate}}^{\text{int}}(\Phi, \delta \Phi) = \sum_{n=1}^{N} \delta d_{in} \cdot \left[ -G_{ij,n-1,n}^{\text{int}} \mu_{jn}^{\text{y}} - G_{ij,n,n+1}^{\text{int}} \mu_{jn+1}^{\text{y}} \right]
\]
(45)

where the gradients \(G_{ij(n)}^{\text{int}}\) assume the form
\[
G_{ij,n-1,n}^{\text{int}} = V_0 \left( \frac{\lambda}{2} d_{in} \otimes d_{j,n-\frac{1}{2}} + \frac{\mu}{2} d_{in} \otimes d_{j,n-\frac{1}{2}} + \frac{\mu}{2} d_{kn} \otimes d_{k,n-\frac{1}{2}} \delta_{ij} + \frac{1}{2} S_{k,n-1,n}^{ij} \right)
\]
\[
G_{ij,n,n+1}^{\text{int}} = V_0 \left( \frac{\lambda}{2} d_{in} \otimes d_{j,n+\frac{1}{2}} + \frac{\mu}{2} d_{in} \otimes d_{j,n+\frac{1}{2}} + \frac{\mu}{2} d_{kn} \otimes d_{k,n+\frac{1}{2}} \delta_{ij} + \frac{1}{2} S_{k,n+1,n}^{ij} \right)
\]
(46)

Gravitational forces. Similar to before, the contributions of the gravitational director forces in the discrete setting amount to
\[
\delta J_{\text{grav}}(\Phi, \delta \Phi) = \delta J_{\text{grav}}^{\text{state}}(\Phi, \delta \Phi) + \delta J_{\text{grav}}^{\text{costate}}(\Phi, \delta \Phi)
\]
(47)

In view of the contribution to the state equations one gets
\[
\delta J_{\text{grav}}^{\text{state}}(\Phi, \delta \Phi) = \sum_{n=1}^{N} \delta \mu_{in}^{\text{y}} \cdot \left[ -f_{j,n-1,n}^{\text{grav}} \right]
\]
(48)

Here, the standard mid-point discretization of (24) yields
\[
f_{j,n-1,n}^{\text{grav}} = 2 A_{n-\frac{1}{2}}^{MP} E_{0}^{ij} d_{j,n-\frac{1}{2}}
\]
(49)

where
\[
A_{n-\frac{1}{2}}^{MP} = \frac{GM}{2 \left( R_{n-\frac{1}{2}}^{MP} \right)^{3}} \left( I - 3 \epsilon_{r_{n-\frac{1}{2}}}^{MP} \otimes \epsilon_{r_{n-\frac{1}{2}}}^{MP} \right)
\]
\[
R_{n-\frac{1}{2}}^{MP} = \left\| \varphi_{n-\frac{1}{2}} - \varphi_{M} \right\| 
\]
\[
E_{r_{n-\frac{1}{2}}}^{MP} = \frac{\varphi_{n-\frac{1}{2}} - \varphi_{M}}{\left\| \varphi_{n-\frac{1}{2}} - \varphi_{M} \right\|}
\]
(50)

The corresponding contributions to the costate equations give
\[
\delta J_{\text{grav}}^{\text{costate}}(\Phi, \delta \Phi) = \sum_{n=1}^{N} \delta d_{in} \cdot \left[ -G_{ij,n-1,n}^{\text{grav}} \mu_{jn}^{\text{y}} - G_{ij,n,n+1}^{\text{grav}} \mu_{jn+1}^{\text{y}} \right]
\]
(51)
Accordingly the gradients $G_{ij}^{\text{grav}}$ assume the form
\[ G_{ij}^{\text{grav}}_{j_{n-1, n}} = A_{n - \frac{1}{2}}^{\text{MP}} E_{j_{0}}^{0} \quad G_{ij}^{\text{grav}}_{j_{n, n+1}} = A_{n + \frac{1}{2}}^{\text{MP}} E_{j_{0}}^{0} \] (52)

**Control forces.** In regard to (25) and due to the fact, that we will deal with a fully-actuated system, $U_{ij}^{ij}$ can be relabeled as control inputs in the optimal control setting. In this context, following contributions to the DNCO in (35)\_1\_3 arise
\[ \delta J_{\text{act}}(\Phi; \delta \Phi) = \delta J_{\text{act}}^{\text{state}}(\Phi; \delta \Phi) + \delta J_{\text{act}}^{\text{costate}}(\Phi; \delta \Phi) + \delta J_{\text{act}}^{\text{control}}(\Phi; \delta \Phi) \] (53)

Once again, we obtain the following contribution to the state equations (35)\_1
\[ \delta J_{\text{act}}^{\text{state}}(\Phi; \delta \Phi) = \sum_{n=1}^{N} \delta \mu_{jn} \cdot f_{n-1, n}^{\text{act}} \] (54)

where the discrete counterpart of (25) yields
\[ f_{n-1, n}^{\text{act}} = U_{n}^{ij} d_{j_{n-\frac{1}{2}}} \] (55)

In view of the contributions to the costate equations (35)\_2 one obtains
\[ \delta J_{\text{act}}^{\text{costate}}(\Phi; \delta \Phi) = \sum_{n=1}^{N} \delta d_{in} \cdot \left[ G_{ij}^{\text{act}}_{j_{n-1, n}} \mu_{jn}^{v} + G_{ij}^{\text{act}}_{j_{n, n+1}} \mu_{jn+1}^{v} \right] \] (56)

Here, the gradients $G_{ij}^{\text{act}}_{j_{n-1, n}}$ assume the form
\[ G_{ij}^{\text{act}}_{j_{n-1, n}} = \frac{1}{2} U_{n}^{ij} I \quad G_{ij}^{\text{act}}_{j_{n, n+1}} = \frac{1}{2} U_{n+1}^{ij} I \] (57)

Eventually we obtain the following contributions to the control equations in (35)\_3
\[ \delta J_{\text{act}}^{\text{control}}(\Phi; \delta \Phi) = \delta U_{ij}^{ij} \cdot \left[ d_{j_{n-\frac{1}{2}}} \cdot \mu_{jn}^{v} \right] \] (58)

**Terminal constraints.** Next, we will elaborate on the contributions of the terminal state constraints $\delta J_{\text{terminal}}$ in (35)\_4 to the DNCO. In regard to the satellite attitude maneuver, the following terminal state constraints $\Psi$ for a rest-to-rest maneuver will be imposed.
\[ \Psi = \begin{bmatrix} \Psi_{i}^{q} \\ \Psi_{i}^{v} \end{bmatrix} = \begin{bmatrix} d_{iN} - \bar{d}_{i} \\ v_{iN} - \bar{v}_{i} \end{bmatrix} \] (59)

Embedding (59) into (35)\_4 eventually leads to
\[ \delta J_{\text{terminal}} = \delta d_{iN} \cdot \delta i_{j} \eta_{i}^{q} + \delta v_{iN} \cdot \delta i_{j} \eta_{i}^{v} + \delta \eta_{i}^{q} \cdot (d_{iN} - \bar{d}_{i}) + \delta \eta_{i}^{v} \cdot (v_{iN} - \bar{v}_{i}) \] (60)

where $\bar{d}_{i}$, $\bar{v}_{i}$ denote the prescribed end-state configuration and velocity respectively.
4 Attitude maneuver of a spacecraft in a circular orbit

The numerical example at hand portrays a satellite performing a rest-to-rest large angle rotational maneuver on a circular orbit while being under the influence of a Newtonian gravitational force field exerted by a spherically symmetric central body. A similar numerical example has been dealt with in [5]. Here, the satellite is modeled as a pseudo-rigid body moving on a prescribed circular path, see also Figure 3 for the general setup. The prescribed path is defined by

$$\varphi = R [\sin \omega t, \cos \omega t, 0]^T$$

where $R$ and $\omega$ are the orbit radius and circular velocity respectively. In this regard all geometrical and material parameters of the system can be found in Table 1. The uncontrolled trajectory for the system at hand is shown in Figure 4. The boundaries for a rest-to-rest maneuver are defined by the initial and terminal rotation tensor

$$R_0 = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad R_N = \exp (\xi \hat{e})$$

where $\hat{e} \in \mathfrak{so}(3)$ is a skew-symmetric tensor with associated axial vector $e = e_3$ and $\xi = \frac{\pi}{2}$ denotes the angle of rotation. Note, that both $R_0 = D_0^I \otimes e_i$ and $R_N = d_N \otimes e_i$ belong to the Special Orthogonal group in 3-space $SO(3)$, such that $R_N^T R_N = I$, $\det R_N = 1$. The time domain for the maneuver is confined to $\Omega = [0, \frac{\pi}{2}]$.

By applying a Newton-Raphson scheme to the DNCO emanating from (35), we obtain the trajectory for the optimally controlled maneuver, see Figure 5. Moreover, a specific component of the director triad for the controlled maneuver is depicted in Figure 6.

5 Conclusion

In the present work the equations governing the motion of an elastic Cosserat point (or pseudo-rigid body) are transcribed into an optimal control problem via a direct approach. To this end, the discrete necessary conditions of optimality have been established for the exemplary attitude control maneuver of a spacecraft in a circular orbit under the influence of a Newtonian gravitational force field. Here, the focus lies on the detailed depiction of the discrete state equations, based on an EM-scheme, and the resulting impact on the discretization of the conditions of optimality.
<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$M$</td>
<td>Mass central body</td>
<td>10</td>
</tr>
<tr>
<td>$m$</td>
<td>Mass satellite</td>
<td>2</td>
</tr>
<tr>
<td>$E_0$</td>
<td>Referential Euler tensor</td>
<td>diag(3, 3, 2)</td>
</tr>
<tr>
<td>$G$</td>
<td>Gravitational constant</td>
<td>10</td>
</tr>
<tr>
<td>$R$</td>
<td>Orbit radius</td>
<td>4</td>
</tr>
<tr>
<td>$\omega$</td>
<td>Circular velocity</td>
<td>1</td>
</tr>
<tr>
<td>$E$</td>
<td>Young’s modulus</td>
<td>$7e3$</td>
</tr>
<tr>
<td>$\nu$</td>
<td>Poisson’s ratio</td>
<td>0.33</td>
</tr>
</tbody>
</table>

Table 1. Parameters (SI units)

Figure 4. Uncontrolled trajectory ($h = 0.01$, $\Omega = [0, \frac{3}{2} \pi]$)

6 Acknowledgment

Support for this research was provided by the Deutsche Forschungsgemeinschaft (DFG) under Grant BE 2285/10-1. This support is gratefully acknowledged.
Figure 5. Controlled maneuver \((N = 100, \Omega = [0, \frac{\pi}{2}\right))\)

Figure 6. Director component \(d_3(3) = d_3 \cdot e_3\)

REFERENCES


Application of optimisation and an artificial neural network to stabilisation of a load relocated under water

Iwona Adamiec-Wójcik, Łukasz Drąg

Faculty of Management and Transport
University of Bielsko-Biała
Willowa 2, 43-309 Bielsko-Biała, Poland
[i.adamiec, ldrag]@ath.bielsko.pl

ABSTRACT

The paper considers a dynamic model of a system consisting of a winch with a pay-load suspended on a rope placed on a moving vessel. The aim of the optimisation problem is the stabilisation of the payload at a given depth despite the sea waves and the vessel motion. The flexible rope is discretised by means of the rigid finite element method. The model allows us to take into account large displacements of the rope due to its flexibility and hydrodynamic forces acting on the submerged part. The nonlinear optimisation methods are applied in order to calculate the rotation angle of the winch such that the load during a defined motion of the vessel remains at the set distance from the undulating seabed. The equations of motion are derived using the Lagrange equations. The drive function is approximated by means of spline functions and the task is solved as a nonlinear optimisation problem in which the equations of motion have to be integrated at each optimisation step. Calculations are carried out for different vessel velocities, mass load and the height of the waves. An artificial neural network is proposed to control the position of the payload.

Keywords: flexible rope, payload positioning, optimisation, rigid finite element method, artificial neural network.

1 INTRODUCTION

Contemporary exploration and exploitation of hydrocarbons from deep water requires highly accurate, efficient and safe offshore engineering installations. To simulate and control motion of vessels, riser and payloads, which frequently occur in such systems, complex dynamics models are used. Modelling dynamics of floating cranes requires consideration of the vessel or platform motion due to the sea environment. Dynamic positioning (DP) is an area of active research which deals with different control methods used in order to ensure either a fixed position or pre-defined path of the floating structure.

One of challenging problems is connected with positioning the payload suspended to an off-shore crane when the load sway is affected by the ship’s motion. A control algorithm based on energy dissipation by using the winch to vary the length of the hoist cable over time is proposed in [1]. However, the influence of water on the rope is not considered. Boundary control of a marine flexible installation system is developed in [2], where the authors propose the adaptive boundary control based on Lapunov’s direct method in order to compensate the rope's vibrations and move the sub-sea payload to the desired position. Heave compensation is especially important when the shipboard crane is moving. The objective of the control strategy presented in [3] is to follow a desired path for the payload despite the heave of the vessel. The system considered consists of a hydraulic-driven winch, a crane-like structure and the rope-suspended load. It is assumed that the crane is a rigid body and the rope with the payload is treated as a spring-mass-damper system. Two methods for determining the drive function of an auxiliary system and the hoisting winch drum in order to stabilise the payload position are presented in [4].

A similar system for which we choose the rotation angle of the winch such that it ensures the realization of a desired trajectory is the subject of this paper. The problem becomes more...
complicated when a part of the rope is submerged and hydrodynamic forces act on the rope. Moreover, the length of the rope changes not only due to winding but also because of its flexibility.

Usually, the methods used for modelling flexibility in offshore structures are those developed for multibody system dynamics such as: finite element method [5,6], lumped mass method [7], segment method [8,9]. A broad review of the methods used is presented in [10]. Here, a special modification of the rigid finite element method [11-13], which enables us to take into account the vibrations of the rope due to hydrodynamic forces and its complex shape, is used.

In the paper a solution of the dynamic optimisation problem is proposed and calculations of the appropriate rotation function of the winch which compensates vibrations of the crane caused by the sea motion are presented. At each optimization step for the set of decisive variables the equations of motion of the system are integrated.

The dynamic optimization is very time consuming and it cannot be applied for control in real-time systems. Thus, in this paper author propose an artificial neural network in order to control the position of the payload. The solutions of the optimization set are used for preparation of the learning set for the neural network.

2 MODEL OF THE SYSTEM

The system considered (Fig.1) consists of a winch placed at the vessel deck, a flexible rope and a payload with mass $m$ suspended at its end (point $A_0$). The position of the payload is adjusted by the drive function $\varphi(t)$ describing rotation of the winch.

![Figure 1](image.png)

**Figure 1.** The system considered consisting of a moving vessel and a winch with a payload

2.1 Discretisation of the rope

The rope is discretised using a special modification of the rigid finite element method. The continuous system is replaced by the system of rigid elements (rfes) connected by massless and non-dimensional spring-damping elements (sdes) as shown in Figure 2a.
The motion of each rfe is described by the following vector of generalised coordinates:

$$\mathbf{q}_i = [x_i \ y_i \ z_i \ \psi_i \ \theta_i \ \varphi_i]^T$$

(1)

where $$x_i, y_i, z_i$$ are coordinates of point $$A_i$$, $$\psi_i, \theta_i, \varphi_i$$ are Euler ZYX angles.

It should be noted, that due to the motion of the winch the rope can reel/unreel on the drum and as a consequence not only the length of the last element changes but also the number of elements into which the rope is divided.

Following the procedure presented in [12] and using homogenous transformations, the kinetic energy of rfe $$i$$ can be presented in the following form:

$$E_i = \frac{1}{2} \text{tr}\{\mathbf{B}_i \mathbf{H}_i \mathbf{B}_i^T\}$$

(2)

where $$\mathbf{B}_i$$ is the homogenous transformation matrix dependent on generalised coordinated, $$\mathbf{H}_i$$ is the constant pseudo-inertial matrix of the $$i$$-th rfe.

The rigid elements form a kinematic chain by means of geometrical constraint equations, which after their double differentiation assume the following form:

$$\ddot{\mathbf{D}}\dot{\mathbf{q}}_i - \ddot{\mathbf{D}}_{i-1}\dot{\mathbf{q}}_{i-1} = \mathbf{G}_i \quad \text{for} \quad i = 1, \ldots, n$$

(3.1)

$$-\ddot{\mathbf{D}}_n \dot{\mathbf{q}}_n = \mathbf{G}_{n+1}$$

(3.2)

where matrix $$\ddot{\mathbf{D}}$$ is constant and matrices $$\mathbf{D}_i, \mathbf{G}_i$$ depend on generalised angles $$\psi_i, \theta_i, \varphi_i$$ and their derivatives.

The choice of generalised coordinates of rfes in the form of (1) means that the elements are treated as unconstrained and thus reactions between elements are introduced by means of generalised forces:

$$\mathbf{Q}_0^F = \mathbf{Q}_0^{(A_0)} = -\ddot{\mathbf{D}}_0^T \mathbf{F}_1$$

(4.1)

$$\mathbf{Q}_i^F = \mathbf{Q}_i^{(A_i)} + \mathbf{Q}_{i+1}^{(A_{i+1})} = \ddot{\mathbf{D}}_i^T \mathbf{F}_i - \ddot{\mathbf{D}}_{i+1}^T \mathbf{F}_{i+1} \quad \text{for} \quad i = 1, \ldots, n$$

(4.2)

where $$\mathbf{F}_i$$ are reactions between rfe $$i-1$$ and rfe $$i$$ presented in Figure 2b.

2.2 Equations of motion

Water environment has to be considered by means of forces acting at the rigid finite elements, which belong to the submerged part of the rope (Fig.3).
Figure 3. Forces acting at rfe $i$: $df_w$ - buoyancy force, $df'_H$ - hydrodynamic drag force, $df_I$ - inertia force

Influence of those forces is taken into account in the equations of motion by means of generalised forces obtained by integration of

$$
\begin{align*}
    df_{w,i} &= f_{0,i}^l \begin{bmatrix} 0 & 0 & 0 \\ g \rho_w C_i & 0 & 0 \\ 0 & g_i \rho_w C_i & 0 \end{bmatrix} d\xi = f_{0,i} \\
    df'_{H,i} &= S_{H,i}(q_i, \dot{q}_i, \xi) d\xi \\
    df_{I,i} &= [C_M C_i \rho_w a_w - (C_M - 1) C_i \rho_w a] d\xi 
\end{align*}
$$

where $\rho_w$ is density of water, $C_i$ is the cross-section area of rfe $i$, $C_M$ is a coefficient, $a_w$ is water acceleration, $a$ is the acceleration of the part of rfe $i$ with length $d\xi$.

Finally, the equations of motion can be written in the following form:

$$
\begin{align*}
    M_0 \ddot{q}_0 + \tilde{D}_0^T \mathbf{F}_1 &= f_0 \\
    M_i \ddot{q}_i - \tilde{D}_i^T \mathbf{F}_{i+1} + \tilde{D}_i^T \mathbf{F}_i &= f_i \quad \text{for } i = 1, \ldots, n - 1 \\
    M_n \ddot{q}_n - \tilde{D}_n^T \mathbf{F}_{n+1} &= f_n 
\end{align*}
$$

where matrix $M_i$ includes mass matrix of element $i$ and of added water, vector $f_i$ includes influence of water environment and moments transferred by spring-damping elements.

The constrain equations remain unchanged.

3 STABILISATION OF THE PAYLOAD

3.1 Nonlinear optimisation

For the optimisation problem considered the model of the system presented in Figure 4 consists of the winch and the rope. It is assumed that vessel motion is known which means that coordinates of point $A_{n+1}$ are known:

$$
\begin{align*}
    x_{A_{n+1}} &= \bar{x}_A(t) \\
    y_{A_{n+1}} &= \bar{y}_A(t) \\
    z_{A_{n+1}} &= \bar{z}_A(t) 
\end{align*}
$$

Because hoisting winch drum may rotate, the rope's length varies, which allows to adjust load position (height $\delta$). Thus, for the optimisation problem the course of drum rotation angle $\varphi(\bar{x}_A(t), \bar{y}_A(t)) = \varphi(t)$ is selected to ensure that despite horizontal movement of the vessel, point $A_0$ will stay at the same distance $\delta$ from the sea bottom (Fig. 4). The vertical displacements of point $A$ is omitted. It can be shown that the vertical movement can be compensated by adoption of:

$$
\varphi = \varphi(\bar{z}_A) + \varphi(\bar{x}_A, \bar{y}_A) 
$$

where $\varphi(\bar{z}_A) = \frac{\bar{z}_A}{R}$. 

769
The objective function is defined as follows:

$$\Omega = \left[ \frac{1}{T} \int_0^T (z_{A_0} + z_{sb} - z(x, y) - \delta)^2 dt \right]^{1/2} \quad (9)$$

where $z(x, y)$ is the function describing sea bottom landform features, $z_{sb}$ is the sea bottom depth for $A_0$ at $t = 0$, $z_{sb} = -(z_{A_0}|_{t=0} - \delta)$.

The function describing the sea bottom landform features is defined in the following form:

$$z(x, y) = b \exp \left( -\left( \frac{x-x_0}{p} \right)^2 - \left( \frac{y-y_0}{q} \right)^2 \right) \quad (10)$$

where $x_0, y_0, p, q$ are parameters of the function describing sea bottom profile.

Calculation of values of functional $\Omega$ involves integration of equations (6) for adopted functions defining $\bar{x}_A(t), \bar{y}_A(t)$. To analyse the problem as non-linear optimisation it is assumed that $\phi_i$ is defined as a spline function of third order. The objective is to search for such values of $\phi_i(i = 1, \ldots, k)$ being the equidistance points of time interval $(0, T)$ which fulfil limitations $\phi_i,_{min} \leq \phi_i \leq \phi_i,_{max}$ and minimize functional $\Omega$ defined in (9).

### Table 1. Rope and water parameters

<table>
<thead>
<tr>
<th>Element</th>
<th>Parameter</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>line</td>
<td>length $L$</td>
<td>300</td>
<td>m</td>
</tr>
<tr>
<td></td>
<td>Young modulus $E$</td>
<td>$10^{11}$</td>
<td>N/m²</td>
</tr>
<tr>
<td></td>
<td>density $\rho$</td>
<td>6500</td>
<td>kg/m³</td>
</tr>
<tr>
<td></td>
<td>diameter $d$</td>
<td>0.07</td>
<td>m</td>
</tr>
<tr>
<td></td>
<td>number of elements $n$</td>
<td>15</td>
<td>–</td>
</tr>
<tr>
<td>water</td>
<td>density $\rho_w$</td>
<td>1025</td>
<td>kg/m³</td>
</tr>
<tr>
<td></td>
<td>$D'_x$ coefficient</td>
<td>0.1</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td>$D'_y$ coefficient</td>
<td>1.0</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td>$C_M$ coefficient</td>
<td>2.0</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td>acceleration $a_w$</td>
<td>$[0 0 0]^T$</td>
<td>m/s²</td>
</tr>
<tr>
<td></td>
<td>distance $\delta$ from the sea bottom</td>
<td>5</td>
<td>m</td>
</tr>
</tbody>
</table>

To solve the problem of non-linear optimisation the downhill simplex method was used. The equations of motion were integrated assuming the step of integration $h \leq 0.1$ s securing proper stability and accuracy of solution.
The data assumed are presented in Table 1. The displacements of the end of the rope was calculated and due to the hydrodynamic resistance they take the course presented in Figure 5. The selection of the angle of rotation of the hoisting winch drum $\phi(t)$ in the problem of stabilization of the rope end at a set height above sea bottom, will concern the compensation of the rope end lifting caused by the vessel movement and elimination of changes caused by set bottom profile.

![Figure 5. Trajectory of point E](image)

It should be noted that when hoisting winch is immobilized line end $E$ is displaced along $z$ axis by approximately 7 m. This results from the fact that rope length is maintained unchanged, despite the effect of hydrodynamic resistance. Total simulation time was $T = 210s$. The interval $(0, T)$ was divided into $k = 14$ subintervals. The vessel moved along $x$ and $y$ axes as shown in Figure 6.

![Figure 6. Vessel movement – point $A_{n+1}$ in $x$ and $y$ directions](image)

It was assumed that the sea bottom profile was flat ($b = 0$ in (10)). Figure 7 presents calculated trajectory of point $E$, before and after optimisation under the assumption that the rope end is kept at a set distance $\delta$ from the sea bottom.
The value of objective function after optimisation was $\Omega = 0.076$, which means that the average deviation of the rope end from set distance $\delta = 5$ m above the sea bottom was below 0.11 m.

The range of angles of the rotation of the hoisting winch drum $\varphi(t)$ selected during optimisation process is presented in Figure 8. The angle of drum rotation was selected to eliminate changes caused by raising the end of the rope due to hydrodynamic resistance and change of sea bottom profile.

The results of dynamic optimisation presented above indicate that through appropriate selection of the hoisting winch drum rotation angle it is possible to control the position of the end of the rope, which will guarantee keeping it at the set level above the sea bottom. The major factor influencing the optimisation process time is the number of decisive variables.
The control in real time of the position of the end of the rope is impossible due to the calculation time of the optimisation problem. The solution can be the use of an artificial neural network.

3.2 Artificial neural network

An artificial neural network (ANN) enables the results of optimisation to be used in real time. In this section we present an application of a sigmoidal ANN with multilayer perceptron (MLP). For the purpose of this section a planar model has been considered with a lumped mass placed at the end of the rope. It is assumed that the motion of point $A$ (vessel) in the direction of axis $x$ is presented in Figure 9a and responds to velocity $v_x(A)$ presented in Figure 9b. Phases of accelerated and decelerated motion last 10 s. The vessel moves with constant velocity (1 m/s, 1.25 m/s and 1.5 m/s) also 10 s.

![Figure 9. Motion of the vessel (point A) a) position b) velocities](image)

Motion of point $A$ in the direction of axis $y$ (vertical) caused by sea waves is assumed to be a harmonic function defined by the formula:

$$y_A(t) = a \cos \omega t$$

where $a$ is the amplitude, $\omega = \frac{2\pi}{T_w}$ is the frequency, $T_w = 10$ s is the wave period.

Different values of the additional lumped mass $m_C$ are also considered. Figure 10 presents exemplary results of the optimization task for the velocity $v_x = \max_t v_x(A) = 1$ m/s and lumped mass $m_C = 0$ with the assumption $a = 0$ (lack of vertical motion) in (11).

![Figure 10. Course of point E before and after optimization](image)

The optimisation task was solved for different velocities and lumped masses. The results obtained were used to formulate learning sets for ANN one hidden layer, the structure of which is presented in Figure 11.
Vertical velocities of the vessel and values of the lumped mass at the end of the rope are inputs for the network. The learning set is created on the basis of the results of the dynamic optimization problem in 441 cases which are combinations of \( v_x \in (1, 1.5) \) in 21 equally spaced points and \( m_c \in (0, 2000) \) also in 21 equally spaced points. When one neuron is assumed to be the output, the following is obtained:

\[
\eta(v_x, m_c) = f\left(\sum_{i=1}^{k} w_i \eta_i\right) + w_0
\]  

where \( w_i \) are neuron weights, \( \eta_i \) is the length of the wound/unwound rope for neuron \( i \) of the hidden layer, \( f \) is the activation function.

In order to estimate the exactness of the approximation the following indicators are used:

\[
E = \frac{1}{u p} \sum_{i=1}^{u} \sum_{j=1}^{p} |\eta_{i,j}^* - \eta_{i,j}| - \text{mean error} \quad (13.1)
\]

\[
E_{\text{max}} = \max_{0 \leq i \leq u, 0 \leq j \leq p} |\eta_{i,j}^* - \eta_{i,j}| - \text{maximal error} \quad (13.2)
\]

where \( u \) is the number of elements in the learning set, \( \eta_{i,j}^* \) is the value calculated by means of ANN, \( \eta_{i,j} \) is the value from the learning set obtained as a solution of the dynamic optimisation task.

Table 2 presents dependence of the errors defined in (13) on the number of neurons in the hidden layer.

<table>
<thead>
<tr>
<th>Type of ANN net</th>
<th>Quantity</th>
<th>Number of ( k ) neurons</th>
</tr>
</thead>
<tbody>
<tr>
<td>MLP 2×16</td>
<td>( E ) [m]</td>
<td>0.0043 0.0015 0.0012 0.0012 0.0012</td>
</tr>
<tr>
<td>MLP 2×16</td>
<td>( E_{\text{max}} ) [m]</td>
<td>0.0825 0.0567 0.0545 0.0533 0.0524</td>
</tr>
</tbody>
</table>

The mean error is about 0.002 m and maximal error \( E_{\text{max}} \) for \( k \geq 10 \) is not larger than 0.09 m.
Figure 12 presents results of ANN for $v_x = 1.22$ m/s and $m_c = 475$ kg for different amplitudes $a$ of the vertical movement of the vessel.

Analysis of the results indicates that even for a high wave, the stabilization of the payload (point $E$) at a depth of 200 m is possible with the deviation not larger than 0.04 m.

4 CONCLUSIONS

The rigid finite element method enables us to formulate an effective model of slender structures used in offshore engineering. Water environment influence by means of buoyancy forces, drag forces, and inertial forces with the added water can be easily considered. The optimisation task gives reliable results which then can be used as training sets for ANN.

REFERENCES


ABSTRACT

The interaction between the mechanical and the control system is crucial for dimensioning and operating a mechatronic system. A typical example for such a system is a horizontal axis wind turbine (HAWT). This paper focuses on the evaluation of a Model Predictive Controller (MPC) for a 3 MW horizontal axis wind turbine designed by W2E Wind to Energy. Within industrial applications in the wind industry, often classical control schemes based on simple PI (proportional-integral) or PID (proportional-integral-derivative) algorithms are used. The main purpose of this type of controller algorithms is safe and efficient operation of the wind turbine. Due to higher requirements by the market regarding the manufacturing costs of wind turbines, the simple controller algorithms are no longer an appropriate way to design a wind turbine. Due to the strong influence of the control system on the mechanical loads acting on a wind turbine, a lightweight design could be reached by using advanced controller schemes, such as MPC, during the development process. A MPC algorithm allows to optimise the wind turbine regarding its overall efficiency, that means minimal manufacturing costs and maximal power output. The presented contribution compares a classical controller, currently used in the turbine and the MPC simulated with a validated multibody model of a wind turbine. As benchmark within this paper, a typical design load case, the 50-year extreme operating gust at cut-out wind speed is considered. It is shown, that the MPC not only reduces the tower top movement significantly but also the output power fluctuation. At the same time the MPC is capable of keeping the rotation speed within its allowable limits.

1 INTRODUCTION

The control development is a very important part within the design process of wind turbines. This design process can be divided into the following six steps:

1. Conception
2. Blade design
3. Control development and preliminary design models
4. Design engineering and strength calculation
5. Construction and erection of the prototype
6. Measurements on the prototype

During the conception, the nominal parameters of the wind turbine, such as nominal power output, rotor diameter, electrical system, and so on, are defined. Secondly, the blades are designed, which includes the aerodynamic as well as the structural concept of the blades. While the aerodynamic concept is determined by the overall conception of the wind turbine, nominal power and rotor radius, the structural concept is mainly influenced by the controller and the preliminary design
models derived in step 3. The preliminary design models, typically based on multibody models, are used to estimate the loads acting on a wind turbine during its life cycle. The steps 2 and 3 depend on each other, so that several iterations are necessary to finalize the blade and the controller design. After finishing the preliminary design phase in step 3 the components of the wind turbine are designed in detail. In step 4 a strength calculation of the components is done in order to verify that they are able to withstand the loads calculated during step 3. When these steps are finished successfully a prototype of the wind turbine is manufactured and erected. This prototype has to prove the predicted properties within a measurement campaign. From this explanation it is evident that the control development has a large impact on the final wind turbine design due to its placement at a very early stage of the design process. In Figure 1 the time line of the wind turbine’s design process and the possibilities and costs of modifications during the development is shown.

![Figure 1](image.png)

**Figure 1.** Possibilities and Costs of modifications during the design process of a wind turbine

As can be seen the control development starts shortly after beginning of the blade design. First calculations are done by the use of a preliminary blade design where the structure of the blade is estimated by a simple beam model. The blade structure, the tower structure and the interface loads of the wind turbine are determined by an iteration between the steps 2 and 3. Due to the strong influence of the controller on the interface loads almost all components are directly or indirectly dimensioned by the control algorithm.

Using advanced controller schemes for load reduction can therefore lead to more efficient wind turbines. Note, that load reduction usually goes along with a reduced power output of the wind turbine. The aim of this work is to show the capabilities of Model Predictive Control to reduce mechanical loads while at the same time aiming for maximum power output. The controller presented is compared to the control scheme currently used. Both controllers are simulated with a highly detailed multibody model. The test case presented in this paper is the operation during a 50 years extreme gust.

## 2 THE WIND TURBINE AND THE MULTIBODY MODEL

The wind turbine this study is based on is the W2E-120/3fc wind turbine, designed and manufactured by W2E Wind to Energy. The prototype of the wind turbine has a rotor diameter of 120 m and a nominal power of 3 MW. The wind turbine is equipped with an innovative drive train consisting of Winergy’s HybridDrive and W2E’s LarusCompact drive train concept. The HybridDrive combines a two stage planetary gearbox with a permanent magnet synchronous generator (PMSG). It’s a medium speed generator controlled by three full converter system. Due to the full converter
system the wind turbine is independent of the grid frequency. The prototype of the 3 MW wind turbine is shown in Figure 2.

![Prototype of the W2E-120/3fc (3 MW wind turbine) erected in Kankel, Mecklenburg-Western Pommerania, September 2013](image)

**Figure 2.** Prototype of the W2E-120/3fc (3 MW wind turbine) erected in Kankel, Mecklenburg-Western Pommerania, September 2013

A preliminary design model of the wind turbine has been built up using the general purpose multi-body program MSC.Adams v2013.1, see [12]. The multibody model interacts with an aerodynamic code and a control system. The aerodynamic loads are simulated using the AeroDyn code v13.00 from the National Renewable Energy Laboratories (NREL) [11]. The simplified interaction scheme of the multibody simulation with the aerodynamic code and the controller of the wind turbine is shown in Figure 3.

![Simplified interaction scheme of the wind turbine model with the aerodynamic code and the controller](image)

**Figure 3.** Simplified interaction scheme of the wind turbine model with the aerodynamic code and the controller

A discrete interface was developed for the interaction of the controller with the multibody program. The aim of this interface is to integrate the same controller software into the multibody simulation as implemented on the programmable logic controller (PLC) of the wind turbine. The interaction scheme represents a software-in-the-loop principle and was developed in analogy to the hardware-in-the-loop principle described in [1, 17]. The multibody program calls the control algorithm at discrete time steps and waits until the controller provides the required data. Due to the fact that
the integrator of the multibody program generally has a variable step size, the interface has to be realised in such a way that the controller is called only at prescribed discrete time steps. On the real PLC of the wind turbine the controller is operated with a cycle time of 10 ms, which is also chosen as prescribed time step for the interface here. As with the real PLC control system the control signals are kept constant between those calls. This interface scheme does not present any real-time capabilities which is, however, not necessary and not realisable for large simulation models.

The multibody model also interacts with the aerodynamic code. This code of NREL calculates the aerodynamic forces based on the blade element momentum theory. For this purpose, the blade is divided into separate aerodynamic elements. As shown in Fig. 3, the multibody code provides the position and velocity of the blade elements. The aerodynamic code provides the aerodynamic forces and moments.

The multibody model in MSC.Adams is parametrically built up. That means, instead of defining the model within the Adams/View preprocessing environment, the model is created within the MATLAB environment. Thereby a MATLAB code is used to generate an Adams command file in the ASCII format, which can be imported by Adams/View. The same principles were also applied for model generation of the high-lift mechanisms of a modern transport aircraft, see [18].

The Adams model comprises a flexible tower model based on a finite element model, blades built up of discrete beams and a detailed drive train model, see [14]. The discrete beams consist of lumped mass elements and Euler-Bernoulli beams. Compared to a blade model consisting of flexible bodies, a higher numerical stability of discrete beams in MSC.Adams during start-up of the wind turbine can be achieved. Furthermore, effects like centrifugal stiffness are taken into account, and the interface loads along the blade can be obtained easily.

As a result, a multibody model with approximately 700 degrees of freedom (DOF) is obtained, see Fig. 4. The simulation of a ten minute time series with turbulent wind conditions on an Intel Core i7-2600 takes about 20 minutes.

Figure 4. Multibody model of the W2E-120/3fc (3 MW wind turbine) using MSC.Adams
3 CLASSICAL CONTROL OF WIND TURBINES

The scope of this section is to give a general introduction into classical control of horizontal axis wind turbines. Due to the high dynamics of wind turbines and their nonlinear behaviour, the control of the mechanical system is a very important aspect for the multibody simulation. The controller strategies can be divided into collective pitch control on the one hand and individual pitch control on the other hand. An overview on collective pitch control for wind turbines, representing classical controller strategies, can be found in [2] and [13]. Besides simple PI (proportional-integral) or PID (proportional-integral-derivative) controller algorithms, also optimal power tracking algorithms are treated. An introduction into individual pitch control for wind turbines, representing advanced controller schemes, is given by [3]. The advanced controller strategies, e.g. Model Predictive Control [9], mainly focus on load reduction of the wind turbine. An extensive overview of advanced controller schemes including active tower damping procedures and controller schemes with wind prediction systems such as light detection and ranging (LIDAR) systems is given within the UPWIND project, see [5]. A field test validation of the advanced controller schemes on two- and three-bladed wind turbines is given by [4].

A typical controller set-up of an industrial wind turbine is shown in Figure 5. This control scheme represents also the current working principle of W2E’s wind turbines. As can be seen, the controlled variable is the generator speed. If it differs from a reference speed the generator torque and the pitch angle are used to minimize this deviation. The main focus of such a scheme is the maximum power output and limit it to its rated value while at the same time guarantee safe operation especially in case of strong changes of the environmental conditions, such as a gust. The loads caused by the control actuation itself are a secondary aspect of these control principles.

In general two different control modes are activated at the wind turbines. The control modes depend on the operation state of the wind turbine. If the wind speed is below rated, also called partial-load region, the wind turbine is mainly controlled via the converters using the generator torque. When the wind speed is above rated the wind turbine is mainly controlled using the blade’s pitch angle. That means according to Figure 5 the lower control path representing the torque control is active only. The detailed torque control principle using a linearised mechanical model of the wind turbine is shown in Figure 6. The wind turbine model includes a rotor model based on aerodynamic characteristics using the dimensionless coefficients for power $c_P$ and for thrust $c_T$ converting the wind speed into the corresponding aerodynamic torque and thrust of the rotor. Due to the use of the aerodynamic characteristics the rotor model is nonlinear and depends not only on the main inflow, but also on the rotational speed of the rotor and the current pitch angle. Furthermore, the wind turbine model contains a linearised mechanical model of the drive train and the tower each. Within Figure 6 the mechanical models are given as simple transfer functions only, so they can be

---

**Figure 5.** Classical control scheme of a wind turbine

---

**Figure 6.** Detailed torque control principle using a linearised mechanical model of the wind turbine.
based on an arbitrary mechanical modelling. Typically the tower is modelled as a clamped single mass spring damper system representing the first eigenfrequency of the tower in thrust direction. The drive train can be modelled by a free three mass rotational springer damper system, which is able to represent beside the rotation of the drive train the first in-plane eigenfrequency of the rotor and the first torsional eigenfrequency of the drive train. Typically the kinematic and dynamic properties of the drive-train are projected with respect to the generator speed. As a result the input values of the drive-train model have to be transferred using the inverse of the overall gearbox ratio \( i_{\text{gear}} \). Finally, the transfer behaviour of the generator is often modelled by first order lag element (PT1). Such a wind turbine model fulfils all requirements as mentioned in [2].

In most cases the control of a wind turbine in the partial-load region (below rated) bases on an optimal power curve, see also [7] and [6]. The optimal power curve is calculated by a theoretical calculation using rigid blades, the ideal aerodynamic profiles of the blades and a constant flow through the rotor disc. The control principle presented in Figure 6 shows that the generator speed \( \omega_{\text{gen}} \) is used as pseudo set point for the torque control. With the current generator speed the optimal power curve provides the corresponding optimal torque \( \tau_{\text{opt}} \) to be applied at the generator. The optimal torque \( \tau_{\text{opt}} \) representing the real set point of the torque control is compared with the actual torque. The offset between actual and optimal control is used within a simple PI scheme to control the wind turbine.

Furthermore, in Figure 6 also a classical control scheme for a wind turbine in the full-load region (at/above rated) is presented. For simplification the control principle using the pitching of the blades is considered only, where the angle of the blades are varied. In industrial applications often the generator torque and the blade pitch are used together to control the wind turbine in the full-load region. Here, the generator torque is kept constant to the nominal torque \( \tau_{\text{gen,sp}} \equiv \tau_{\text{nom}} \). Within the full-load region the aim of the control is the safe operation of the wind turbine. Therefore the controller tries to keep the generator speed on a constant level. The difference between nominal and actual generator speed is used in combination with a PID scheme to control the wind turbine. Due to the fact that the pitch drive reacts much slower than the converter-generator-combination larger oscillations around the nominal values occur.

Overall, there are two different controller schemes for the partial- and full-load region. Whereas the aim of the control scheme in the partial-load region is the maximum power output in the full-load region the generator speed is kept constant to the nominal value. Load reduction for several components is a secondary aim of the control scheme. Often, other control schemes are added to the two main control schemes mentioned above, e.g. for reduction of drive train oscillations. A problem occurs due to the fact that the control loops can influence each other leading to an unexpected behaviour of the complete system such as self-excited vibrations. To overcome this problem an advanced control scheme for wind turbines will be proposed in the next section.

**4 MODEL PREDICTIVE WIND TURBINE CONTROL**

Alternatively to classical control concepts as described before Model Predictive Control (MPC) can be used to operate a wind turbine. In contrast to classic single-input-single-output control concepts here the different control aims, such as maximum power and minimal mechanical stress, as well as multiple control signals and saturations can be handled within one controller. The control signal is derived by optimizing an user defined cost-function which includes the computation of a simplified wind turbine model. The next section introduces the principle of MPC before the required real-time capable wind turbine model is presented and compared to the multibody model introduced before.

**4.1 Principle of Model Predictive Control**

Compared to the classic control loops shown in Figure 5 the MPC based control scheme in Figure 7 only features one single control block. The controller inputs are the reference signals (trajectories),
Figure 6. Detailed control system of a wind turbine including a linearised mechanical model and a rotor model based on characteristic diagrams.
the limitations as well as the state vector of the wind turbine controlled. The state vector is thereby reconstructed from available measurements by using for instance a Kalman Filter. Within the MPC the optimal control signals $u(\cdot|k)$ is derived by an optimization of the form

$$u(\cdot|k)_{opt} = \arg\min_{u(\cdot|k)} J(u(\cdot|k))$$

with $J$ being a quadratic cost-function of the form

$$J = \sum_{i=0}^{H_P} \left( y(k+i|k) - r(k+i|k) \right)^2 Q(i) + \sum_{i=0}^{H_U-1} \left( \Delta u(k+i|k) \right)^2 R(i)$$

Within this cost-function the deviation of the future output $y(\cdot|k)$ of the wind turbine, predicted by computing a simplified model, and the desired reference $r(\cdot|k)$ as well as control signal changes $\Delta u(\cdot|k)$ are penalized. As illustrated in figure 8 the plant output is evaluated within the prediction horizon $H_P$, while the control signal can only be changed within the control horizon $H_U$. A more comprehensive explanation on MPC can be found e.g. in [10].

4.2 Control Model

In this subsection the control model used within the MPC is briefly introduced. Other than the multibody simulation model the DOF of the control model is as small as possible. To do so the tower is modelled as simple spring-damper system of the form

$$f_{\text{thrust}} = m_t \ddot{x}_t + d_t \dot{x}_t + k_t x_t,$$

where $f_{\text{thrust}}$, $m_t$, $d_t$ and $k_t$ are the thrust force, the nacelle mass, the tower damping and stiffness, respectively. The coordinate of the tower top displacement is $x_t$. The drive train with its two inputs aerodynamic rotor torque and generator torque $\tau_{\text{rotor}}$ and $\tau_{\text{gen}}$, respectively, is modelled as two lumped masses representing rotor inertia $I_r$ and generator inertia $I_g$ connected via a spring-damper-system. This system can be described as

---

**Figure 7.** Block diagram of a MPC controlled wind turbine.
\[ \tau_{\text{rotor}} = I\ddot{\phi}_{\text{r}} + d_{\text{r}} \Delta \dot{\phi}_{\text{g}} + k_{\text{r}} \Delta \phi_{\text{g}}, \]
\[ -\tau_{\text{gen}}/i_{\text{gear}} = I_{\text{g}} \ddot{\phi}_{\text{g}} - d_{\text{g}} \Delta \dot{\phi}_{\text{r}} - k_{\text{g}} \Delta \phi_{\text{r}} , \]
where \( i_{\text{gear}} \) is gearbox ratio, \( d_{\text{r}} \) and \( k_{\text{r}} \) are drive train damping and stiffness, respectively. The rotor and generator accelerations are \( \ddot{\phi}_{\text{r}} \) and \( \ddot{\phi}_{\text{g}} \). The difference of rotor and generator azimuth angle and speed are defined as
\[ \Delta \phi_{\text{rg}} = \phi_{\text{r}} - \phi_{\text{g}}/i_{\text{gear}} \quad \text{and} \quad \Delta \dot{\phi}_{\text{rg}} = \dot{\phi}_{\text{r}} - \dot{\phi}_{\text{g}}/i_{\text{gear}}. \]

Here, \( \phi_{\text{r}} \) and \( \phi_{\text{g}} \) are identical to the rotor speed \( \omega_{\text{rotor}} \) and generator speed \( \omega_{\text{gen}} \), respectively. The aerodynamic inputs thrust force and rotor torque are generally defined by
\[ f_{\text{thrust}} = 0.5 \rho \pi R^2 v^2 c_T(\vartheta, \omega_{\text{rotor}}, v) \quad \text{and} \]
\[ \tau_{\text{rotor}} = 0.5 \rho \pi R^2 v^3 \omega_{\text{rotor}}^{-1} c_P(\vartheta, \omega_{\text{rotor}}, v). \]
Here, \( \rho \) and \( R \) represent the air density and the nominal rotor radius, respectively. The relative inflow on the rotor is obtained by \( v = v_{\text{W}} - v_{\text{rotor}} \). The coefficients \( c_T \) and \( c_P \) are given as three dimensional characteristic grids. Those equations combined with an additional state for the pitch angle \( \vartheta \) result in a 6th order state space system with the state vector
\[ x = [\dot{\phi}_{\text{r}}, \dot{\phi}_{\text{g}}, \Delta \phi_{\text{rg}}, x_{\text{t}}, \dot{x}_{\text{t}}, \vartheta]^T. \]

The controlled outputs \( y \) of the system are generator speed \( \dot{\phi}_{\text{g}} \), tower top velocity \( \dot{x}_{\text{t}} \) and electrical output power \( P_{\text{el}} \),
\[ y = [\dot{\phi}_{\text{g}}, \dot{x}_{\text{t}}, P_{\text{el}}]^T. \]

The control inputs of this system are the generator torque \( \tau_{\text{gen}} \) and the pitch angle velocity \( \dot{\vartheta} \) so that the input vector \( u \) is
\[ u = [\tau_{\text{gen}}, \dot{\vartheta}]^T. \]

The MPC used for this study is based upon linear, time discrete state space models of the form
\[ x_{k+1} = Ax_k + Bu_k \quad \text{and} \]
\[ y_k = Cx_k + Du_k, \]
so that the model introduced before is discretised and linearised at a given operation point.
4.3 Model Comparison

In this section the controller model introduced above is compared to the multibody model of section 2. For comparison a wind step from 10 m/s\(^{-1}\) to 14 m/s\(^{-1}\) is applied to the MBS-Model. Thereby the classic controller is still activated, so that pitch angle and generator torque are varied. The same input signals wind speed, pitch angle and generator torque are fed into the simplified control model. The resulting output and state variables are plotted in Figure 9. The control actions result in slightly different results for the multibody and the control model. The generator speed shows equal dynamic behaviour, but an offset for steady state operation. This is caused by differences in the aerodynamic model, which lead to a different aerodynamic rotor torque. The tower acceleration shows good agreement directly after the wind step. But the control actions taken afterwards result in different dynamic behaviour. As the results will show, this difference is tolerable for the controller. Just as the generator speed the tower top displacement shows an offset in steady state operation due to differences in the aerodynamic model.

Note that during operation the model errors which lead to constant offsets are handled and eliminated by the Extended Kalman-Filter used for state estimation [16] (see Figure 7) which is not part of this paper.

![Figure 9. Comparison of reduced controller model and MBS Model](image)

5 RESULTS

In this section classic and MPC controlled wind turbine operation are compared. For the 50 year extreme operation gust (abbreviation EOG50) occurring close to cut-out wind speed is one of the
crucial operating conditions for wind turbine design and certification it is used for comparison in this section. In Figure 10 the wind turbine reaction on a 50-year extreme operation gust which increases the wind speed from 25 m/s to up to 35 m/s is plotted.

The comparison of the two control concepts reveals that throughout the whole time the MPC can maintain the electrical output power within a narrow band of 0.5 MW, while it fluctuates with the classic controller by 1.0 MW. At the same time the MPC is capable of reducing the tower top acceleration significantly and hence minimize the mechanical stress of the wind turbine during this extreme event.

The basis for that result is, that the MPC receives it’s state vector $x$ from the EKF which also estimates the mean wind speed (see also [8, 15]). This information allows the MPC to react better in this situation. For occurring loads during gust situation depend on the rotor position the simulation was conducted with initial rotor angles of 0°, 30°, 60° and 90°. It was found, that the results are similar. Since input loads to the whole system are the blade root bending moments, which vary cyclic the peak-to-peak values were compared for the different azimuth angles. Thereby it was found, that the peak-to-peak values of the blade root bending moments in flapwise direction are reduced by up to 8.5% and by 6% in average compared to the operation with classical control. The edgewise bending moments are reduced by up to 17% and 16% in average. Since the pitch angle at cut-out wind speed is high, the reduction of the edgewise bending moments has a significant influence on the reduced tower movement observed.
6 CONCLUSIONS AND OUTLOOK

In this study the capability of Model Predictive Control in load reduction for wind turbines was evaluated and presented. A highly sophisticated and validated multibody model of a wind turbine was presented and used for testing the capabilities of the MPC. The results show that even with a significantly reduced control model within the MPC it is possible to reduce occurring loads e.g. in extreme situations. Since those scenarios are design drivers and therefore determining the amount of required material, the reduction of those loads is of high importance and as shown in this paper, can be achieved by the usage of MPC.

Next steps involve improving the robustness of the MPC and the extension towards partial-load region as well as the implementation on a PLC for real-time testing, first in a hardware-in-the-loop environment and afterwards in the field.

Acknowledgement

The authors would like to thank the German Federal Ministry for Economic Affairs and Energy for supporting this project (KF3357201CL4).

REFERENCES


Energy Optimal Manipulation of an Industrial Robot
ECCOMAS Thematic Conference on Multibody Dynamics 2015

†Thomas Lauß† *, 2Peter Leitner*, 3Stefan Oberpeilsteiner† *, 4Wolfgang Steiner† *

†Faculty of Engineering and Environmental Sciences
University of Applied Science Upper Austria
Stelzhamerstraße 23, 4600 Wels, Austria

† Institute for Mechanics and Mechatronics
Vienna University of Technology
Getreidemarkt 9, 1060 Wien, Austria

ABSTRACT
The main goal of this contribution is to determine the excitation of an industrial robot, such that the energy consumption becomes a minimum during the manipulation of the tool center point (TCP) from a start position to a given end point within a predefined time. Such tasks can be restated as optimization problems where the functional to be minimized consists of the endpoint error and a measure for the energy. The gradient of this functional can be calculated by solving a linear differential equation, called the adjoint system [3, 5, 4]. On the one hand the minimum of the cost functional can be achieved by the method of steepest descent where a proper step size has to be found or on the other hand by a Quasi-Newton algorithm where the inverse of the Hessian can be appreciated.

Keywords: optimal control, multibody dynamics, adjoint system, optimization, calculus of variation.

1 INTRODUCTION
Nowadays it is impossible to imagine industrial facilities without robots. Many of these manipulators have to do one operation repeatedly. Therefore it is obvious that the movements should be planned such that the energy effort is as little as possible. Due to the high amount of robots in the industry and their nearly unlimited operating time, already a small reduction of the energy per manipulation has an enormous consequence on the total industrial energy consumption.

In this contribution an approach to such inverse dynamical problems is presented. It starts from an optimal control formulation of the problem by introducing a cost functional which has to be minimized subject to a differential equation as an additional condition (c.f. [3, 4]). The gradient computation of the cost functional is based on the so called adjoint equations. Due to better convergence a Quasi-Newton method is used instead of the simple gradient method to find a minimum of the cost functional. Therefore the Hessian matrix is approximated by using the BFGS-algorithm (c.f. [10]).

For an energy optimal manipulation of the robot two different formulations of the cost functional are tested. On the one hand, the quadratic consideration of the inputs and on the other hand the mechanical power is taken into account. A scrap-function is used in both formulations which is important to reach the given end point.

The identified movements were tested on a PUMA six axis robot. With the measured control variables and the generalized velocities the required energy was evaluated. Based on this test data a considerably energy reduction was detected.

1Scientific Assistant, E-Mail: thomas.lauss@fh-wels.at, Phone: +43 (0)50804-4498
2Scientific Assistant, E-Mail: peter.leitner@fh-wels.at
3Scientific Assistant, E-Mail: stefan.oberpeilsteiner@fh-wels.at, Phone: +43 (0)50804-4488
4Professor, E-Mail: wolfgang.steiner@fh-wels.at, Phone: +43 (0)50804-43230
2 PROBLEM DEFINITION

At first, let us consider a nonlinear dynamical system

\[ \dot{q} = v, \]
\[ M(q)\dot{v} = F(q, v, u, t), \]

(1)

where \( q \in \mathbb{R}^n \) is the vector of generalized coordinates and \( v \in \mathbb{R}^n \) is the vector of generalized velocities. In addition, \( M \) is the \( n \times n \) mass matrix and \( F \in \mathbb{R}^n \) the force vector. The vector \( u \) indicates the control variables in an opened or enclosed region \( \Omega \subseteq \mathbb{R}^m \). By introducing the vector of state variables \( x^T = (q, v)^T \) we may rewrite Equation (1) by

\[ \dot{x} = f(x, u, t), \quad x(t_0) = x_0. \]

(2)

In general the force vector \( f \) is a continuous vector field which depends on the states \( x \), controls \( u \) and on time \( t \). In robotics \( u \) are usually torques in the joints which appear linear on the right side of Equation (2). For this special case the equations of motion result in

\[ \dot{x} = a(x, t) + B(x, t)u(t) \]

where \( B \) is the \( 2n \times m \) state- and time-variant input matrix. If the number of degrees of freedom \( n \) and the number of controls \( m \) are equal the system is called \textit{fully-actuated}. For the case that the number of controls \( m \) is less than the number of degrees of freedom \( n \) we call the system \textit{under-actuated}.

In robotics the position and velocity of the tool center point (TCP) will be of particular interest instead of the joint angles and angular velocities. Hence, the system output \( y \in \mathbb{R}^l \) is given by

\[ y = g(x). \]

In order to meet a predefined end point we have to satisfy the boundary condition

\[ g(x(t_f)) = \bar{y}. \]

(3)

However, we substitute the boundary condition of Equation (3) by the optimal control problem

\[ \dot{x} = f(x, u, t) \]
\[ J = \int_{t_0}^{t_f} h(x, u, t) \, dt + S(t_f, x(t_f)) \rightarrow \min \]

(4)

where the integral describes the energy consumption and the \textit{scrap function} \( S \) measures the end point error. If the closed region \( \Omega \) is not empty the solution of the \textit{optimal control} problem of Equation (4) leads to an energy optimal manipulation of the dynamical system of Equation (2).

From Pontryagin’s minimum principle (c.f. [2]) the controls and states which lead to the optimal solution can be obtained. Therefore a two-point boundary value problem has to be solved, which is derived from the \textit{Hamiltonian}

\[ H(x, p, u, t) = h(x, u, t) + p^T f(x, u, t) \]

(5)

where \( p(t) \in \mathbb{R}^n \) is a continuous and piecewise continuously differentiable function which denotes the \textit{costate or adjoint} variables. In every region \( t \in [t_0, t_f] \) where \( u^*(t) \) is continuous the minimum condition

\[ H(x^*, p, u^*, t) = \min_{u \in \Omega} H(x^*, p, u, t) \]

(6)

as well as the \textit{adjoint} (costate) equation

\[ \dot{p} = -H_x(x^*, p, u^*, t) \]

(7)
must hold. In Equation (7) the term $H_x$ is a vector of partial derivatives of $H$ with respect to the components of the state $x$. In addition at the endpoint $t_f$ the transversality condition

$$ p(t_f) = S_x(g(x(t_f))) $$

must be taken into account. Finally, the solution of Equation (4) is equivalent to the boundary value problem

$$ \dot{x} = H_p \quad x(t_0) = x_0 $$
$$ \dot{p} = -H_x \quad p(t_f) = S_x(g(x(t_f))) $$
$$ u^* = \arg \min_u H $$

(8)

where the term $H_p$ is a vector of partial derivatives of $H$ with respect to the components of the co-state $p$. Hence, if the controls are torques in a mechanical system, the Hamiltonian of Equation (5) are linear in $u$. As a consequence the control $u$ does not appear in $H_u$. On the one hand the control $u$ becomes infinite, if the region $\Omega$ is not bounded and on the other hand the singular case can occur. If this is the case, the control cannot be eliminated from Equation (8) and so it leads to a differential-algebraic boundary value problem which is considerably harder to solve.

### 3 GRADIENT COMPUTATION

To determine the gradient of the cost functional (4), Equation (2) is added to the cost functional

$$ J = \int_{t_0}^{t_f} h(x,u,t) + p^T (f(x,u,t) - \dot{x}) \, dt + S(t_f,x(t_f)). $$

(9)

The Lagrange-multipliers $p$ are denoted as adjoint variables and are arbitrary at this point. Integration by parts of the term $\int p \dot{x}$ leads to

$$ J = \int_{t_0}^{t_f} (h + p^T f + \dot{p}^T x) \, dt + S(t_f,x(t_f)) - p^T x |_{t_0}^{t_f} $$
$$ = \int_{t_0}^{t_f} (H + p^T x) \, dt + S(t_f,x(t_f)) - p^T x |_{t_0}^{t_f}. $$

(10)

In order to find a minimum of the cost functional $J$ with respect to $u$ we consider the variation of $J$ according to a small change $\delta u$ which is given by

$$ \delta J = \int_{t_0}^{t_f} \left[ (H_x^T + \dot{p}) \delta x + H_u^T \delta u \right] \, dt + \left[ S_x^T (t_f,x(t_f)) - p^T (t_f) \right] \delta x(t_f) + p^T (t_0) \delta x(t_0). $$

(11)

Due to the fact that no variation of the states at $t = t_0$ is allowed, the term $p(t_0) \delta x(t_0)$ is zero. If the adjoint variables are defined, such that

$$ \dot{p} = -H_x \quad \text{and} \quad p(t_f) = S_x(t_f,x(t_f)). $$

(12)

the variation of $J$ according to Equation (11) is reduced to

$$ \delta J = \int_{t_0}^{t_f} H_u^T \delta u \, dt. $$

(13)

Equation (12) may be solved backwards in time starting at $t = t_f$ after the system equations have been solved forward in the time interval $t \in [t_0,t_f]$. The largest possible increase of $\delta J$ is obtained, if $\delta u(t)$ is chosen in the direction of $H_u^T$. For that reason $H_u^T$ may be considered as the gradient of the functional $J(u)$. 

791
4 NUMERICAL DETERMINATION OF THE OPTIMAL CONTROL

Based on the adjoint gradient computation outlined in the previous section we may now search for a control \( u \) which minimizes the objective functional \( J \). First of all, the method of steepest descent is described, where we always walk a certain distance along the negative gradient until we end up in a local minimum of \( J \). Due to the costly line search step during every iteration and the slow convergence the gradient method is extended to a Quasi-Newton method. Therefore we solve the problem of finding \( u \) such that the gradient becomes zero.

4.1 The Method of Steepest Descent

The method of steepest descent tries to find a minimum of a function or subsequently of a functional by walking always along the direction of its negative gradient. This concept has first been developed to optimal control problems by H.J. Kelley [8] and A.E. Bryson [9].

The gradient is already derived from the adjoint system which is shown in Section 3. Now we use \( H_u^T \) and simply walk a short distance along the negative gradient of \( J \). By reason of numerics the continuous functions are discretised. So the cost functional reads

\[
J(u) \approx \hat{J}(u_1, u_2, \ldots, u_N)
\]

where \( u_i = u(t_i) \) and \( t_1, \ldots, t_N \) is a sequence of consecutive time steps in the interval \([t_0, t_f] \). A variation of the controls \( u_i \) leads to a variation of the cost functional

\[
\delta \hat{J} = \sum_{i=1}^{N} \frac{\partial \hat{J}}{\partial u_i} \delta u_i.
\]

On the other hand, the variation \( \delta \hat{J} \) can be expressed by Equation (13) which, after discretisation, results in

\[
\delta \hat{J} = \sum_{i=1}^{N} H_{u,i} T \Delta t_i \delta u_i
\]

where \( H_{u,i} \) is the evaluation of \( H_u^T \) at \( t = t_i \). Hence, the gradient of the discretised functional may be identified as

\[
\frac{\partial \hat{J}}{\partial u_i} = H_{u,i} T \Delta t_i
\]

with \( \Delta t_i = t_i - t_{i-1} \). For walking in the direction of the negative gradient a small number \( \kappa > 0 \) has to be chosen to get the increment

\[
\delta u_i = -\kappa H_{u,i} T \Delta t_i.
\]

If \( \kappa \) is sufficiently small, the updated control \( u_i + \delta u_i \) will always reduce the cost functional \( J \). However, finding the number \( \kappa \) such that \( J \) is absolutely reduced may require several simulations of the system equations. For that purpose, the increments given by Equation (15) are considered as functions of \( \kappa \). After solving the equations of motion with \( u + \delta u \) as inputs also the objective function \( J \) becomes ultimately a function of \( \kappa \). By means of a line search algorithm one may find a number \( \kappa \) in a predefined interval \([0, \kappa_{\text{max}}] \) which minimizes \( J \).

4.2 Application of a Quasi-Newton Method

It is well known that the convergence of the gradient method is rather slow. Hence, a Newton method provides an alternative approach to find the minimum of the cost functional \( J \). The basic idea is the following one: If \( \hat{u} = (u_1^T, u_2^T, \ldots, u_N^T)^T \), the minimizing vector \( \hat{u} \) is defined by a zero gradient, i.e. by the equations

\[
\nabla J = \begin{bmatrix} \frac{\partial J}{\partial u_1}, \cdots, \frac{\partial J}{\partial u_N} \end{bmatrix}^T = 0
\]
which can be solved for $\bar{u}$ by Newton’s method. However, the Hessian $H = \left( \nabla \bar{j} \right)_\bar{u}$ is required for that purpose. To avoid the full computation of $H$, which would be extremely time consuming, several quasi-Newton methods have been developed. They all approximate the Hessian by using the gradients of successive Newton-iterations. For example, the Hessian can be estimated efficiently by the well known Broyden-Fletcher-Goldfarb-Shanno (BFGS)-Algorithm (c.f. [10]). Even its inverse can be efficiently obtained by applying the Sherman-Morrison formula (c.f. [11]).

We compute an approximation $\tilde{H}^{-1}$ of the inverse of the Hessian from the BFGS-algorithm. Then, an increment $\delta \bar{u}$ of the discretised control signal is given by

$$
\begin{pmatrix}
\delta u_1 \\
\delta u_2 \\
\vdots \\
\delta u_N 
\end{pmatrix} = -\tilde{H}^{-1} \nabla \bar{j} 
$$

Note, that it is strongly recommended to use a quasi-Newton method which directly approximates the inverse of the Hessian. Otherwise, if the original Hessian is computed, a very large and dense matrix must be inverted, since the number of components of $J$ might become large.

The inverse of the Hessian after $k+1$ iterations is given by

$$
\tilde{H}_{k+1}^{-1} = \left( I - p_k q_k^T \right) \tilde{H}_k^{-1} \left( I - q_k p_k^T \right) + p_k p_k^T 
$$

where $I$ is the identity matrix, $p_k$ is the gradient direction of the $k$th-iteration and $q_k$ is the change of the gradient during the last iteration [11].

<table>
<thead>
<tr>
<th>method</th>
<th>number of Iterations</th>
<th>$f(x^*)$</th>
<th>$| \nabla f(x^*) |$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gradient method</td>
<td>57</td>
<td>0.123</td>
<td>1.198</td>
</tr>
<tr>
<td>Quasi-Newton method (BFGS)</td>
<td>23</td>
<td>$5.4 \cdot 10^{-12}$</td>
<td>$9.2 \cdot 10^{-6}$</td>
</tr>
</tbody>
</table>

In Figure 1 and Figure 2 the comparison of the convergence of the gradient and the Quasi-Newton method is pictured. Therefore the function $\text{fminunc}$ from the MATLAB-Optimization toolbox is used. The function which has to be minimized is called Rosenbrock’s banana function. It is a non-convex function which is often used as a performance test for optimization algorithms. In Figure 1, it is conspicuous that the convergence of the gradient method is very bad near the
minimum. However the Quasi-Newton method is considerably better due to the use of the Hessian matrix. In Table 1 the comparison of the two methods is shown.

5 APPLICATION TO THE SIX-AXIS-ROBOT

The presented method is used to optimize the energy consumption of the robot with respect to two different definitions of the cost functional. The reason why we have chosen this PUMA robot is that a lot of different parameters are available which are necessary for the evaluation and verification of the results. Afterwards the simulation results are verified at a real six-axis-robot which is shown in Figure 4.

5.1 Problem definition

The system consists of three degrees of freedom, $\theta_1$, $\theta_2$ and $\theta_3$ which denote the relative rotation angles of the joints. Due to the complicated structure of the equations of motion and the minor influence on the energy consumption the three wrist joints are fixed. First of all the equations of motions are derived and have the form

$$ \dot{x} = f(x, u, t), \quad x(t_0) = x_0 $$

where $u = [M_1, M_2, M_3]^T$ contains the torques of the motors and $x = [\theta_1, \theta_2, \theta_3, \dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3]^T$ is the vector of states of the dynamical system. The system output

$$ y = g(x) $$

is a nonlinear function which depends on the states and describes the coordinates of the tool center point $y = [x(t), y(t), z(t)]^T$.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Start Position</th>
<th>Final Position</th>
<th>Start Velocity</th>
<th>Final Velocity</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\theta_1$</td>
<td>$0^\circ$</td>
<td>$-90^\circ$</td>
<td>$0 \text{ rad/s}$</td>
<td>$0 \text{ rad/s}$</td>
</tr>
<tr>
<td>$\theta_2$</td>
<td>$0^\circ$</td>
<td>$-10^\circ$</td>
<td>$0 \text{ rad/s}$</td>
<td>$0 \text{ rad/s}$</td>
</tr>
<tr>
<td>$\theta_3$</td>
<td>$0^\circ$</td>
<td>$45^\circ$</td>
<td>$0 \text{ rad/s}$</td>
<td>$0 \text{ rad/s}$</td>
</tr>
<tr>
<td>$x_{TCP}$</td>
<td>$-0.15320 \text{ m}$</td>
<td>$0.81441 \text{ m}$</td>
<td>$0 \text{ m/s}$</td>
<td>$0 \text{ m/s}$</td>
</tr>
<tr>
<td>$y_{TCP}$</td>
<td>$0.92112 \text{ m}$</td>
<td>$-0.15320 \text{ m}$</td>
<td>$0 \text{ m/s}$</td>
<td>$0 \text{ m/s}$</td>
</tr>
<tr>
<td>$z_{TCP}$</td>
<td>$0.02032 \text{ m}$</td>
<td>$0.22233 \text{ m}$</td>
<td>$0 \text{ m/s}$</td>
<td>$0 \text{ m/s}$</td>
</tr>
</tbody>
</table>

The PUMA (Programmable Universal Machine for Assembly, or Programmable Universal Manipulation Arm) is an industrial robot arm developed by Victor Scheinman for General Motors at the Stanford University.
For the energy optimal manipulation of the robot from a start-point \( x_0 \) to a given end-point \( \bar{y}, \dot{\bar{y}} \) (c.f. Table 2) within a predefined time \( t_f \) two different definitions of the cost functional are tested. First, a very common approach in pertinent literature that minimizes the quadratic signal energy (c.f. [6]) is applied. In this case the cost functional is given by

\[
J_1 = \int_{t_0}^{t_f} uu^T u \, dt + S(t_f, x(t_f)).
\]  
(18)

The advantage of this method is that the convergence rate of the optimization process is quite good due to the quadratical formulation. On the other hand this measure is not of practical importance, it is rather a mathematical construct which has been established due to good convergence. Hence, it suggests itself that one could take the real mechanical energy into account. In this case the cost functional

\[
J_2 = \int_{t_0}^{t_f} uu^T F_u v \, dt + S(t_f, x(t_f))
\]  
(19)

has to be minimized, where \( v \) is the velocity of the appropriate degree of freedom (DOF) and \( F_u \) combines the input signals with the proper DOF. The integrand of Equation (19) symbolizes the mechanical power at one point in time.

The scrap-function \( S \) of Equation (18),(19) describes the endpoint error and is specified by

\[
S(x,t) = \alpha \begin{cases} 
\beta \left[ y(x) - \bar{y}\right]^2 + \left[ \frac{\partial y}{\partial q} - \dot{\bar{y}} \right]^2 
\end{cases}
\]  
(20)

where \( \alpha \) and \( \beta \) are proper weighting factors and \( \bar{y}, \dot{\bar{y}} \) contains the position and velocity of the endpoint in coordinates of the system output.

5.2 Results

The left diagram of Figure 5 shows the signal energy effort of the standard manipulation in comparison to the optimization with respect to the signal energy. For the sake of completeness also the signal energy expenditure of the optimization with respect to the mechanical energy is pictured. As a result the reduction of the signal energy after the optimization process is about 47% with respect to the standard manipulation of the robot control. On the right hand side of Figure 5 the real mechanical energy effort is pictured. A reduction of 34% could be achieved if the mechanical energy is taken into account in the cost functional. Hence, the minimal signal energy does not ensure an energy optimal manipulation of the robot.

![Figure 5. comparison of the energy expenditure](image-url)
In Figure 6 the joint angles of the optimized motions in comparison to the standard manipulation of the robot are plotted over time. It can be seen that the prescribed end position of the motion, optimized with respect to the mechanical energy, is not met exactly. This angular deviation results in a small endpoint error with a magnitude of 2.49% with respect to the TCP-vector \([x(t_f), y(t_f), z(t_f)]^T\) at the end-time \(t_f\) which is marked in Figure 3. In practical terms this means that one could reduce the energy expenditure by 34% if the end position is modified slightly.

In consideration of the fact that we do not take friction into account, the comparison of the measure versus the simulation is quite good. The trend of the energy expenditure over the time during the manipulation is shown in Figure 7. The solid line results from a real measurement of the robot which is pictured in Figure 4 while the dashed line shows the energy consumption of the simulation.

The energy conservation for the measured signal amounts to 0.84Ws while the simulated results reduces the energy about 2.5Ws. Both values differ clearly from each other because we do not take friction into account.

6 CONCLUSIONS AND OUTLOOK

There are two main statements summing up the results. On the one hand this paper should reveal that the trajectory with minimal signal energy does not lead automatically to the energy optimal manipulation of the robot. Nevertheless in practice such quadratic input terms are often taken into account because this leads to less stress of the motors. In simply terms you can say that the electrical parts are more protected against overheating and the operation life span is increased additionally if the torques remain small.

On the other hand the effort of the mechanical energy results from a small endpoint error. Due to the fact that the mathematical model of the robot is conservative, no dissipation of the energy
is possible. In other words, the energy consumption is independent of the specific motion and depends only on the initial and final state of the robot. Certainly this begs the question about the purpose of this paper. Nevertheless some practical applications exists. A typical example for this field of application would be the planing of a production line. First the positions of the robots are determined such that the robot can reach every essential location. Only a small change of the basic position of the robot reduces the energy effort of the manipulation.

If we consider the functional of Equation (19) in detail one can recognize that the integrand can be a negative value for example if the velocity is positive and the torque is negative. That means that the motor decelerates the motion and recuperates energy which reduces the total consumption. Nowadays modern robots are able to recuperate energy during a manipulation. However, if older robots are used which do not have the ability to recuperate energy during braking the cost functional can be adapted such that this effects are considered.

A video can be found at https://ycutu.be/i2K2znfmE8I where the standard manipulation is overlaid with the optimized motions. The small endpoint error for the mechanical energy optimal manipulation can be seen very well in the video.

For the results in Section 5.2 we neglected the three degrees of freedom of the wrist and fixed them to keep the equations of motion and the necessary matrices simple. However, if we consider this joint angles in the system equations it is possible to reach a predefined endpoint in different ways. This means that more than one final configuration of the robot exists which meet the end point in the coordinates of the tool center point. Therefore this algorithm leads to the solution with the lowest potential energy. In other words the movements of the heavy and large bodies which require most of the energy are as small as possible.

For such cases the optimization process can be implemented in the robot control. Furthermore the identification can be done during operation. Instead of the forward simulation the measures of the previous manipulation can be used to solve the adjoint system and calculate the gradient. Hence, the required energy decreases during the manipulations of the robot. A big advantage is that it is not necessary to exchange any part of the robot, only an update of the robot control is required.

**ACKNOWLEDGEMENTS**

This project was supported by the program "Regionale Wettbewerbsfähigkeit ÖÖ 2010-2013", which is financed by the European Regional Development Fund and the government of Upper Austria.
REFERENCES


Mechatronic solution of components cooperation in the device for gait reeducation

Slawomir Duda*, Grzegorz Gembalczyk*, Jacek Jurkojc#, Slawomir Kciuk*, Robert Michnik#

*Faculty of Mechanical Engineering, Silesian University of Technology, Akademicka 2A, 44-100 Gliwice, Poland

#Faculty of Biomedical Engineering, Silesian University of Technology, Akademicka 2A, 44-100 Gliwice, Poland
[Jacek.Jurojc, Robert.Michnik]@polsl.pl

ABSTRACT

Effective re-education of patient's walk is largely dependent on the repeatability of the conducted exercises. Striving to guarantee the patients an intensive training in sufficient time, rehabilitation centers use various kinds of devices supporting the work of physical therapists. A mechatronic device dedicated to the re-education of walk has been developed in the Department of Theoretical and Applied Mechanics of the Silesian University of Technology. The work presents a general operating concept of the system and an experimental verification of the assumed control mechanism for the case of walk in unloaded conditions. The device is a MIMO (Multiple-Input Multiple-Output) system. Control system processes the measuring signal from the sensors: relief force, rope deviation angle, a sensor detecting the pressure with which the foot touches the floor and encoders recording the position of the relief system. Acquisition of all recorded signals, both digital and analogue, is done in a specially designed interface for signal conditioning. Drive motors work by adjusting the rotation speed. The device operations are controlled by a computer equipped with the RT-DAC4/PCI card for real-time operations. MATLAB/Simulink software is used to program the device controlling system. The computer is directly connected to the signal conditioning. Algorithms implemented in the control system of the drives use the PID controller. The values of setpoints for drive controllers responsible for the movement of an individual device axis were set based on the optimization of the timeline. Response of the drive system to the requested forces with the use of a device numeric model was investigated. During the optimization of parameters of the algorithm responsible for the following movement, a rope deviation angle and for the weight compensation system a deviation of the relief force from the requested value were minimized.

Keywords: mechatronic device, rehabilitation, real-time operations.

1 INTRODUCTION

In the modern world, the number of traffic accidents and civilization-related diseases leading to the loss of motor abilities is constantly growing [2, 5]. Numerous diseases and injuries may permanently affect this ability while hindering ordinary functioning of a person and often preventing one from performing daily activities that seem trivial to those in possession of full motor skills. Such a situation makes the patient dependent on the support of other persons and prevents autonomous performance of physiological activities or social functions. In case of occurrence of walking disorders, rehabilitation of the patient becomes a necessity. It is aimed at the recovery – as far as possible – of the lost motor abilities. This is related to retraining the patient's ability to walk in new circumstances [3, 4, 6, 12, 14].
The modern approach to rehabilitation assumes that the conducted physical therapy sessions are to lead to a full or partial recovery of functions lost in the course of the illness or disease. In case of patients with reduced mobility, the recovery of motor abilities is one of the basic tasks of the rehabilitation. Standard walk re-education methods are based on guiding and ensuring the patient's safety by a physiotherapist. The application of mechatronic devices supporting the movements of the segments of lower limbs (such as Lokomat) or allowing for walk in unloaded conditions (for example, Gait Trainer, ZeroG) may constitute an alternative to these standard methods [13].

Additionally, the movement of the lower limb in these devices is limited only to the sagittal plane, without ensuring proper movements in the transverse planes. It is also not without significance that the movement on the treadmill is not the same as moving in space. Treadmill training introduces habits different from those assumed to be correct in walking. All this may lead to the strengthening of improper movements, which may cause arthritis or lead to future traumatic injuries.

As a result of works conducted in the Department of Theoretical and Applied Mechanics of the Silesian University of Technology, a device for the re-education of motor functions has been developed. The device has been equipped with a real-time drive control system. The completed project is innovative, as the conducted domestic and worldwide market analysis has demonstrated that no such device has been used in the rehabilitation of persons having problems with locomotion [7]. A device that would connect the current findings in the fields of biomechanics and mechatronics. The application of the device in rehabilitation of persons with locomotion disorders, with underlying conditions of both traumatic and neurological nature, provides an improvement of the working conditions of the staff conducting the rehabilitation. Also the patient's safety is improved, as they will be able to perform walking-related exercises while being protected against fall and to perform the exercises in pre-set unloading conditions. An additional advantage over similar devices available in the market is the possibility to learn to climb the stairs and to cross obstacles that may be encountered in everyday life [1].

2 CHARACTERISTICS OF THE DEVICE

The structure of the device is based on the structure of an overhead travelling crane. As a result of installing three drives, the movement of the sling of the person in rehabilitation is possible in all axes of the Cartesian coordinate system. Fig. 1 presents the arrangement of the individual axes.

![Figure 1. Directions and senses of the axes of the global coordinate system corresponding to the movements of individual drives.](image-url)
The basic mechanical part of the device is constituted by drive systems responsible for the transfer of the movement from the motor to the working units.

The transmission of control and measurement signals between the walking re-education device and the computer with the control system is conducted by means of two RT-DAC4/PCI real time boards connected with a signal conditioning interface. A control unit equipped with servo inverters of each of the motors is an additional element necessary for the correct operation of the device. As in each of the drives speed control is used, the servo inverter works as a regulator controlling the torque of the motor so as to ensure the required rotational speed. Each of the control signals is thus first sent to the control unit which works out the settings for the drive motors [9, 10].

Analogue signals from the remaining sensors installed in the device are sent directly to the board conditioning the analogue signals. These sensors include the constructed devices for rope force measurement and the measurement of the sling rope deflection angle (fig. 3) [11]. The data transmission diagram has been presented in figures 4.

The control of the device is conducted by means of the MATLAB/Simulink suite. Two tabs are visualized in the main window of the application. Each of the tabs is responsible for the processing of different measurement, control and servomotor control signals. Due to the real-time library included in the MATLAB/Simulink software suite, the user of the programme may freely operate these signals [16].

3 DRIVE CONTROL

The algorithm controlling the device’s drives, encompassing the X, Y and Z axes, has been divided into three independent systems due to the expected performance method and the requirements of movement of the individual sub-assemblies. The control is performed in a closed control system using PD regulators by generating proper voltages which are supplied to BLDC motors.
A follow-up control system has been implemented for the X and Y axes. In this case, the algorithm of operation performs a certain wave of the controlled value, while the wave is not known. The purpose of this system is to control the object in such a way so as the changes of the regulated value would directly follow the changes of the pre-set value. In this case, the system aims at the minimization of the rope deflection angle. It is also possible to set a trajectory for the movement of the rope suspension point in the XY axis of the device [8, 15].

For the Z axis – the dynamic compensator of rope length, an algorithm aimed at keeping a constant force value in the rope has been implemented. This is necessary to achieve a constant unloading value required in some of the rehabilitation and walking re-education processes.

In the proposed follow-up movement control system in the X and Y axes, the proper value of the rotational speed control signal of the drive motor is selected based on the misalignment signal. The misalignment signal equals the value of the crane rope deflection from the perpendicular (the pre-set rope deflection angle value equals zero). This angle is dependent on the coordinates of the crane trolley and the location of the patient. The idea behind the algorithm controlling the follow-up movement of the crane has been presented in Fig. 6.

**Figure 3.** Structural scheme of the drive system in mechatronic device for locomotor training.

![Figure 3](image)

**Figure 4.** Block diagram of the regulation system

<table>
<thead>
<tr>
<th>X axis</th>
<th>Y axis</th>
<th>Z axis</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLDC motor</td>
<td>Drive system</td>
<td>Actuator (girder + hoist)</td>
</tr>
<tr>
<td>BLDC motor</td>
<td>Drive system</td>
<td>Actuator (trolley + hoist)</td>
</tr>
<tr>
<td>BLDC motor</td>
<td>Drive system</td>
<td>Actuator (hoist)</td>
</tr>
</tbody>
</table>

4 **Verification algorithm control**

As a result of the conducted numerical simulations regarding the developed models of the complex human-machine-control system, PID controller settings have been indicated for the follow-up mode of the operation of the device with unloading. In this operation mode, the task of the device is to follow the moving person to provide fall protection while allowing for setting any value of the unloading force. The tests of the correctness of the system operation have been conducted with the participation of a healthy person moving in a space limited by the movement of the crane sling (fig. 5).
During tests using the APAS movement analysis system, the movement of the centre of the subject’s mass was established. The movement of the subject has been registered using three digital video cameras manufactured by Basler with a sampling frequency of 200 Hz. Reflective markers have been placed on anthropometric points on the body of the subject, which allowed to determine the trajectory of the movement of the centre of mass. The tests have been conducted during walk without unloading, during walk with the application of an algorithm controlling the X-axis movement (modes 1-3) and controlling the movement in X and Y axes (modes 4-6). In modes 1 and 4, unloading at the level of 20% has been applied. In modes 2 and 5, the unloading was 40% and it amounted to approximately 50% in modes 3 and 6. The mean speed of walk was 0.25 m/s.

The displacements of the centres of mass established in the tests in each of the modes were used to determine the displacement ranges in relation to the individual axes (fig. 1). In each of the calculation modes, a lower range of displacements in relation to the vertical axis and a higher range in relation to the horizontal axis were noted as compared to the mode without unloading. In modes 4-6 (with control in relation to the horizontal axis), the range of displacements in relation to the vertical axis was higher than in the case of modes 1-3. The range of displacements in relation to the lateral axis in modes 4-6, however, was lower than in the case of modes 1-3.

![Figure 5. Video camera image with measurement point marked](image)

**Figure 5.** Video camera image with measurement point marked

![Figure 6. The range of displacements of the centre of mass in relation to a) the vertical axis b) the lateral axis](chart)

**Figure 6.** The range of displacements of the centre of mass in relation to a) the vertical axis b) the lateral axis
5 CONCLUSIONS

The work presents an innovative device used to support the rehabilitation process of persons affected by a walking impairment. The use of a computer as the management unit allows for the modification of algorithms and parameters of the control system, which may be motivated by the occurrence of new factors that were not considered earlier. Moreover, the continuous recording of the registered parameters obtained from the installed sensors was made possible, thus allowing for the assessment of rehabilitation progress.

The algorithms developed for the control of the device allow to conduct rehabilitation in two basic modes – as a protective follow-up system and in unloading conditions. Both modes were analyzed and tested, exhibiting the correctness of the proposed control algorithms.

The application of PID controllers in the control of the drives, despite their simplicity, allowed to decrease the excessive displacements in relation to the lateral axis by 13% at mean and to increase to range of movement in relation to the vertical axis. This confirms the efficiency of the applied control algorithm ensuring a greater reflection of the correct walk.

References


Nonlinear dynamics and bifurcation of a vibration absorber-harvester system

K. Kecik∗, P. Brzeski†, P. Perlikowski‡#

∗ Mechanical Engineering Faculty
Lublin University of Technology
Nadbystrzycka 36, 20-618 Lublin, Poland
k.kecik@pollub.pl

† Division of Dynamics
Lodz University of Technology
Stefanowskiego 1/15, 90-924 Lodz, Poland
piotr.brzeski@p.lodz.pl

‡ Department of Civil & Environmental Engineering
National University of Singapore
Engineering Drive 2, Singapore 117576
przemyslaw.perlikowski@p.lodz.pl

ABSTRACT
This paper examines a novel concept and design for simultaneous energy harvesting and vibration mitigation using the same host structure. The system consists of a main body (oscillator) and an absorber (pendulum). Additionally, the oscillator’s suspension is an active construction, composed of the magnetorheological damper (MRD) and the shape memory alloy (SMA) spring. The main aim of this work is analysis of the smart elements influence on the energy harvesting and mitigation of oscillations. The obtained results show that the MR damper and SMA spring can modify the dynamics of the systems. This results in increase or decrease of harvested/absorbed energy. Additionally, an influence of active elements on the stability of solutions is shown.

Keywords: Energy Harvesting, Magnetorheological Damper, SMA Spring, Control, Mitigation of Vibrations.

1 INTRODUCTION
Recently, energy harvesting (EH) from vibrating systems has gained immense popularity. The harvested power depends on the amount of accessible kinetic energy and the efficiency of the harvester. Generally, vibratory energy harvesting can be classified according to the mechanism by which energy is converted into electricity. The most popular are [1]: piezoelectric energy harvesting, in which recovered electricity is a result from pressure, which is the direct result of the piezoelectric effect (under mechanical strain, piezoelectric material produces internal electrical charge), electrostatic energy harvesting, energy harvesting depends on changing the capacitance of vibration-dependent varactors, magnetostrictive energy harvesting, in which applying a mechanical stress changes their domain magnetization, electromagnetic energy harvesting, in which electricity is a results of electromagnetic induction (Faraday’s law). In this work we apply electromagnetic energy harvesting because of it is easy and cheap to design. Disadvantages of such device is that has scalability issues and low energy density because they usually require magnets and coils.

Most harvesters reported in the literature are linear vibration resonator and operate at several tens of Hertz, or even more than 100Hz. Only, in some special cases, such as human, see or wind motions, frequency of the vibration sources is lower than 10Hz. It has been shown that under certain conditions, the presence of non-linearity in the device may cause an improvement of performance as compared to the standard linear systems [2]. Energy harvesters whose are described by nonlinear equations of motion can potentially scavenge energy over a broader frequency range as compared to the linear harvesters. The parametric resonance can occur in parametrically excited systems, when the parametric frequency is near twice the fundamental frequency. This phenomenon can be used to harvest more vibration energy.
The parametric pendulum is the system of great interest for many years, because of its complex dynamical behavior. The concept of using mechanical pendula systems for energy extraction has been given a considerable attention in recent years. Most of the work in this area has been conducted in the Centre for Applied Dynamics Research (CADR) by group of Prof. Wiercigrzech at the University of Aberdeen. Their idea based on the conversion of the kinetic energy of sea waves into rotational motion of the pendulum, mounted on the floating pontoon. In the paper [3] Authors show a numerical and experimental study of a such system with energy extraction. Next, Horton et al. [4] proposed a method for parameter identification of a pendulum experimental pendulum rig for rotational motion. A theoretical analysis on the maximum (optimum) recovered energy from rotation of the parametric pendulum has been conducted by Nandakumar at al. [5]. They propose a system in which a control and energy dissipation have been represented in terms of additional torque (added to the equation). Positive values of the control torque represents energy input, while negative denotes energy output. Interestingly method to utilizing parametrical resonance for harvesting energy from low-frequency vibrations has been investigated by Tian et al. in [2]. A pendulum-type architecture with electromagnetic induction as the energy conversion mechanism has been proposed. The prototype device with a natural frequency close to 2Hz has been tested experimentally. In recent years excessive analytical, numerical and experimental studies of the rotational motion of the parametric pendulum have been conducted in [6, 7, 8].

On other hand, the pendulum can used as the dynamical vibration absorber. The dynamical absorbers include important class of pendulum absorbers: an autoparametric vibration absorbers. Autoparametric resonance occurs when the vibration of the primary system act as parametric excitation of the second system. The autoparametric vibration pendulum absorber is designed to absorb energy from the primary system (main mass). This absorption effect is efficient only in the limited band of vibration frequencies of the main system [9]. Unlike the classical absorber, the use of the pendulum absorber does not result in excitation of vibrations with considerable amplitudes at other frequencies. The influence of smarts elements as SMA spring and MRD to control dynamics is presented in [10]. Author shows influence of MRD and SMA spring on absorption effect and stability of periodic solution. Additionally, simple control swinging of the pendulum, based on an activation of smart elements is tested (numerically and experimentally). Liao et al. [11] presented an active-adaptive tuned vibration absorber based on magnetorheological elastomer. They study two feedback types of activation force and incorporated phase-lead compensator to eliminate the time delay effect during signal processing. Mirsanei et al. [12] analyze an adaptive tuned dynamic vibration absorber based on the smart slider-crank mechanism.

The conception of simultaneously vibration suppression and energy harvesting can be found in the literature. Kecik and Borowiec [13] proposed a an autoparametric pendulum system to energy harvesting. They discuss energy harvesting effectiveness due to different kinds of pendulum motion. The obtained results show that, the chaotic motion of the pendulum results in the highest voltage value. Hassaan [14] propose a novel concept for using the mass-spring vibration absorber as the energy harvester. He define a new frequency called harvesting frequency depending on the mass ratio of the absorber and independent of the main system damping.

The aim of this work is to carry out a fundamental study on the use of a single device (the pendulum attached to the oscillator) to both suppress vibration and harvest energy simultaneously. Usually, the harvester devices are mounted in the pendulum pivot construction, therefore control by the pendulum’s damping is difficult. To improve effectiveness of energy harvesting (or absorption effect), the smarts elements in the oscillator’ suspension are introduced. This active suspension allows control of the dynamics and EH. This absorber-harvester dynamic system can be used both rotational motion (dedicated to sea waves harvesters) or swinging (vibration suppression devices) of the pendulum.
2 DESCRIPTION OF VIBRATIONAL ABSORBER-HARVESTER

2.1 Absorber-harvester system

In Fig.1 a scheme of the absorber-harvester system is presented. The pendulum is made of a rod (length $l$ and mass $m_2$) with small ball of mass $m_3$ at its end. It is attached to a main body of mass $m_1$ (oscillator). The oscillator is suspended on the spring made of SMA and MRD. The system is excited by the linear spring (with stiffness $k_2$), which the end executes harmonic motion $y(t) = Q\cos(\omega t)$, where $Q$ is amplitude and $\omega$ is frequency of excitation, respectively. The har-

![Figure 1. Physical model of absorber-harvester system and equivalent circuit.](image)

vester is mounted in the pendulum pivot. The energy converter consists of a rotor (a cylindrical permanent magnet) and a stator, composed of a two identical windings fixed to housing (orthogonally positioned), connected to the same resistive electrical loads ($R_L$). We assume that, electrical load is pure resistive. The $R_W, L_W$ denote the resistance and inductance of the winding respectively. When, the magnet (the pendulum) rotates or swings, then electromagnetic force $F_{em}$ is generated.

2.2 Electrical circuit equation

The electrical power generated from the rotatory harvester is dissipated across a load resistor $R_L$. Applying Kirchoff’s voltage law to the electrical circuit yield

$$L \dot{i} + (R_L + R_W)i = \kappa \dot{\phi}.$$  \hspace{1cm} (1)

The parameter $\kappa$ denotes transduction factor characterizes the magnetic flux density of two windings, and can be calculated according to [2]:

$$\kappa = N k_w B_{max} l_m \rho,$$  \hspace{1cm} (2)

where $N$ is the total number of turns of each winding, $k_w$ the winding coefficient, $l_m$ the length of the magnet, $B_{max}$ is the maximum magnetic flux density at the centroid of the winding in the circumferential direction and $\rho$ is the radius of magnet. The unit of parameter $\kappa$ is $[Tm^2]$ (or $[Vs]$). Assuming, that the effect of the leakage inductance of the winding is negligible at low frequencies the eq. (1) can be simplified to [15, 16]

$$(R_L + R_W)i = \kappa \dot{\phi}.$$  \hspace{1cm} (3)
The electromagnetic force $F_{em}$ couples the mechanical and the electrical systems and has the same value, but opposite oriented:

$$F_{me} = k_i, \quad (4)$$

$$F_{em} = c_e \dot{\phi}. \quad (5)$$

The parameter $c_e$ describes the electric damping of the rotatory magnet. Based on eq. (3) and eqs. (4) and (5), the electrical damping coefficient yield

$$c_e = \frac{N^2 k_1^2 B_m^2 f_m^2 B^2}{R_L + R_W}. \quad (6)$$

While, the recovered voltage can be estimated from Ohm’s law

$$U = iR_L = \phi \frac{Nk_B B_m l \rho R_L}{R_L + R_W}. \quad (7)$$

The averaged power is calculated from the root mean square (RMS) value of the generated voltage

$$P = \langle \frac{U_{RMS}^2}{R_L} \rangle. \quad (8)$$

The average power is express in using the angular velocity of the pendulum $\dot{\phi}$. 

### 2.3 Models of smart elements

The MRD is described by hyperbolic tangential function proposed in paper [17]. This model depends on the displacement and velocity of MRD piston

$$F_{MR}(\dot{x}, x) = c \dot{x} + d \tanh(e_1 \dot{x} + e_2 x), \quad (9)$$

where $d$ is the friction force coefficient related to the rheological behaviour, produced by MR fluid, $\dot{x}$ is the piston velocity of MRD. The parameters $e_1$ and $e_2$ are positive constant determining the shape of the hysteresis loop. The $c$ denotes viscous damping coefficient. The considered MRD model consists of a combination of the viscous damping and a Coulomb friction.

To describe behaviour of SMA spring we applied Falk approach [18]. In this approach free-energy function based on the analogy between SMA uniaxial stress-strain curves and the electric field magnetization curves of ferromagnetic materials. Applying classical definition of stress-strain, and introducing the restoring force and the displacement of the oscillator yield

$$F_{SMA}(T, x) = a_1(T - T_M)x - a_2 x^3 + \frac{a_2^2 x^5}{4a_1(T_A - T_M)}. \quad (10)$$

The parameters $a_1, a_2$ are positive constants describing material properties. The great advantage of Falk model is its simplicity. Note, that this model becomes linear for $a_2 = 0$. The detailed information about derivation of both models is in papers [10, 17].

### 2.4 Equations of motion of absorber harvester system

Differential equations of motion of the system presented in Fig. 1 are obtained by application of Lagrange equations of the second kind:

$$(m_1 + m_2 + m_3) \ddot{x} + c \dot{x} + d \tanh(e_1 \dot{x} + e_2 x) + a_1(T - T_M)x - a_2 x^3 + \frac{a_2^2 x^5}{4a_1(T_A - T_M)} + \left(m_2 + \frac{1}{2} m_3\right) l \left(\ddot{\phi} \sin \phi + \dot{\phi}^2 \cos \phi\right) = k_2 Q \cos \omega t, \quad (11)$$
\[
\left( m_2 + \frac{1}{3}m_3 \right)I^2 \ddot{\phi} + c_\phi \dot{\phi} + \left( m_2 + \frac{1}{2}m_3 \right)I(\ddot{x} + g) \sin \phi + F_m = 0. \tag{12}
\]

Equation 11 describes the oscillator motion, while eq.12 determining the pendulum dynamics. Note, that both equations are coupled by inertial terms.

The equations of motion are transformed in dimensionless form by introducing the dimensionless time \( \tau = \omega_0 t \), where \( \omega_0 \) is the natural frequency of the oscillator with SMA spring and \( X = x/x_{st} \), where \( x_{st} \) is a static displacement of SMA spring in ambient temperature:

\[
\ddot{X} + \alpha_1 \dot{X} + \alpha_3 \tanh(\delta_1 \dot{X} + \delta_2 X) + (\theta - 1)X - \beta_1 X^3 + \beta_2 X^5 + \mu \lambda (\dot{\phi} \sin \phi + \dot{\phi}^2 \cos \phi) = q \cos \vartheta \tau,
\]

\[
\dot{\phi} + (\alpha_2 + \xi) \dot{\phi} + \lambda (\ddot{X} + 1) \sin \phi = 0. \tag{14}
\]

The natural frequency of the oscillator with SMA spring is calculated for martensite phase and \( x_{st} \) where \( \tau \) time. The equations of motion are typical for an autoparametric systems. The electrical damping plays role of a gain of the mechanical damping.

The dimensionless parameters are as follow:

\[
\begin{align*}
\alpha_1 &= \frac{c}{M \omega_0}, & \alpha_2 &= \frac{c_\phi}{(m_2 + m_3)^2 \omega_0}, & \alpha_3 &= \frac{d}{M \omega_0^3 x_{st}}, & \vartheta &= \frac{\omega_1}{\omega_0}, & \omega_1^2 &= \frac{a_1 T_M}{M} x_{st} = \frac{Mg}{a_1 T_M}, & \tau &= \omega_0 t \\
\theta &= \frac{T}{T_M}, & \beta_1 &= \frac{T_A}{T_M}, & \delta_1 &= e_1 x_{st} \omega_0, & \delta_2 &= e_2 x_{st}, & \beta_2 &= \frac{a_2 x_{st}^2}{\omega_0^3 M}, & \mu &= \frac{(m_2 + m_3)^2}{M x_{st}^2}, \\
M &= \sum_{i=1}^{3} m_i \lambda &= \frac{(m_2 + m_3) x_{st}}{(m_2 + m_3)^2}, & q &= \frac{k Q}{M x_{st} \omega_0^3}, & \xi &= \frac{c_e}{(m_2 + m_3/3) \omega_0}, \phi &= \frac{\phi_d}{\phi_d} = \frac{\phi}{\omega_0}.
\end{align*}
\]

Note, for \( \theta = 2 \) and \( \beta_1 = 0 \), the SMA spring model became a linear. The obtained equations of motion are typical for an autoparametric systems. The electrical damping plays role of a gain of the mechanical damping.

### 3 NUMERICAL RESULTS

#### 3.1 Absorption effect under smart suspension

Considering, that the absorber-harvester system can be designed as the dynamical absorber, it is necessary to understand the influence of smart elements on this phenomenon. Usually, the smart elements are mounted in different applications in order to change dynamics (during motion). All calculations have been performed using software for numerical continuation, Auto07p [19, 20] and Matlab package. The results, that have been obtained by continuation give us an overview of system dynamics demonstrating all possible periodic solutions. The values of parameters have been taken from the real laboratory rig. The parameters describing the mechanical systems are: \( \alpha_1 = 0.1, \alpha_2 = 0.069, \xi = 0.001, \lambda = 0.25, \mu = 6, q = 0.35, \beta_1 = 0.1, \beta_2 = 0.025, \delta_1 = 100 \) and \( \delta_2 = 1 \). The electrical parameters characterizes prototype of the harvester (mounted and tested on the laboratory rig) and getting from literature [2]: \( \kappa = 0.1328 T m^2, R_L = 140 \Omega, R_W = 55 \Omega \).

First, the position in parameters space of absorption region should be found, hence we compute the resonance curves. The response frequency for the oscillator and the pendulum for a system with MRD and SMA spring are shown in Fig.2. The case of response of classical suspension (without MRD and SMA spring) is marked by black line. The unstable solutions are marked by dash-dotted line (black or red), while the solid line denotes stable periodic solutions. Note, that pendulum swinging for \( \vartheta \approx 0.79 - 1.33 \). However, the vibration absorption phenomenon takes place \( \vartheta \approx 0.82 - 1.17 \), while the best region is located near the resonance peak (for frequency \( \vartheta = 0.95 \)). Interestingly, that after crossing the \( \vartheta = 1.17 \), an amplitude of the oscillator nontrivial (NT) solution (the pendulum and oscillator vibrates) is higher then semi-trivial (ST) solution (oscillator vibrates, only).
Figure 2. Frequency response for the oscillator (a) and the pendulum (b), for \( \alpha_3 = 0.1, \theta = 1.5, \beta_1 = 0.1, \beta_2 = 0.025 \).

We can see, that MRD (green line) reduced the amplitude and frequency range of the pendulum operates. The resonance curve for system with SMA spring (red line), changes shape (bends to the right side). Additionally, the resonance region is shifted in left side. The amplitude of the pendulum depends on the frequency of excitation. Note, for frequency nearly \( \vartheta = 0.7 \) a new stable and unstable solutions appears. Unfortunately, for low frequencies, the ST solution has a higher amplitude, what is dangerous for the real systems.

It is shown, that with a proper choice of suspension parameters, the angular velocity of the pendulum can increase (then increases \( \alpha \), too). The highest angular velocity obtained for system with the SMA spring (Fig.2b). Moreover, the smarts elements can be used to elimination or move of the unstable regions. However, an active elements (especially SMA spring) can introduce a new bifurcations: branch point (BP), period doubling (PD), limit point (LP) or Neimark-Sacker (NS), what much complicates the dynamics.

Figure 3. Bifurcation diagrams: \( \alpha_3 \) versus maximal amplitude of the oscillator (a) and the pendulum (b), for \( \theta = 2, \beta_1 = \beta_2 = 0, \) and frequency \( \vartheta = 0.95 \).
The influences of the MRD ($\alpha_3$) on the absorption region (Fig.3a) and the angular pendulum velocity (Fig.3b) are shown. The bifurcation diagrams are made by one parameter continuation for fixed frequency of excitation which is lowest, $\vartheta = 0.95$ (absorption effect is most effective). Analyzing the results in both diagrams, we can conclude that MRD practically, not decreases vibration of the oscillator’s amplitude. The reduced amplitude has similar value about 0.4, what means that effectiveness has similar level. However, the reduced oscillator vibration ration between ST and NT is decreased. Moreover, the MRD reduces the angular pendulum velocity. The critical value of MR damping, for which the pendulum stopped, equals to $\alpha_3 = 0.25$ (Fig.3b).

The influence of SMA spring on amplitude of the main system (Fig.4a) and the pendulum velocity (Fig.4b) are shown. The absorption region is located for $1 < \theta < 3$. Interestingly, that reduced amplitude has similar value about $X_{\text{max}} = 0.4$ (Fig.4a), while the angular pendulum velocity has different value from 0.3 to 0.5 (Fig.4b). This means, that SMA spring can be used to energy recovery without loss of absorption reduction.

However, for low temperature ($\theta < 1$), the stable solution can change in unstable with the high amplitude. The new bifurcations points appear.

In Figs.5a and 5b two parameters space plots are presented. The first diagram shows averaged recovered power under MR damping influence. The diagram is made for fixed initial conditions: $X = \dot{X} = \phi = 0$ and $\varphi = 0.1$. The most effective EH region takes place for $\vartheta \approx 0.85$ (entry in resonance) and $\approx 1.25$ (exit from resonance). The maximum averaged power reaches a value of 30 milliwatts, while the critical MR damping equals about $\alpha_3 = 0.26$ and good corresponded to the bifurcation diagram presented in Fig.3a. Note, that in the absorption region (for $\vartheta = 0.95$), the recovered power equals about 10 milliwatts. For the frequency of $\vartheta = 1$, the energy is recovered in large range of temperature. Obtained results, show that the highest EH occurs on the boundary between ST and NT solutions.

Figure 5b shows influence of SMA temperature on recovered energy. This diagram is much complicated, and resonance region is expanded. The higher EH is located for low temperature (martensite phases) and reaches value of 35 milliwatts. The increase of temperature can improve or deteriorate of power, depending on the frequency of excitation. The high level of recovered energy can be produced by rotational or chaotic motion [13].
3.2 Pendulum’s rotation under smart suspension

The presented absorber-harvester system can be applied energy from rotational motion of the pendulum (classical rotatory harvester). However, the smart suspension can changing dynamics in unexpected way [10]. Therefore, in this section this problem of rotational under an active suspension is analyzed. To receive rotational motion of the pendulum, the amplitude of excitation increased to \( q = 2.5 \) and modified the pendulum’s parameter to: \( \lambda = 0.12, \mu = 17.3 \).

In Fig.6a the bifurcation diagrams: frequency-angular pendulum velocity is presented. This plot is made for typical suspension (viscous MRD and SMA spring). The find all possible the pendulum’s rotation, the continuation method is used. The calculations started from the periodic solution (point denoted as no.1), in the direction indicated by the arrow. The dashed-dot line denotes unstable, while continuous (or red) line stable solutions. In the stable region pendulum executes regular rotation.

Analyzing the results, we can conclude that existing three zones for rotation: (I)- \( \vartheta \approx 0.8 \), (II)-about \( \vartheta \approx 1 \), (III)- about \( \vartheta \approx 1.9 \).

![Figure 5](image1.png)

Figure 5. Two-parameter plots: frequency-MR damping-averaged power (a), frequency-temperature of SMA-averaged power (b). The electrical parameters are: \( \kappa = 0.1328Tm^2, R_L = 140\Omega, R_W = 55\Omega \).

![Figure 6](image2.png)

Figure 6. Bifurcation diagrams: \( \vartheta \) versus maximal angular pendulum’s velocity, for \( \theta = 2, \beta_1 = \beta_2 = 0, \alpha_3 = 0(a) \) and \( \alpha_3 = 0.15(b) \).
The better region for EH is (III), in which angular velocities are higher. Note, that this diagram consist of two separately bifurcation paths which characterizes positive (anti-clockwise direction) or negative (clockwise direction) rotation of the pendulum.

The results from the continuation method are verified by classical integration and basins of attraction (Fig.7b). In this diagram two small rotations regions and one large in which the pendulum is in equilibrium coexist. The pink colour is responsible for positive, while gray negative rotation. The large area (azure colour)) denotes ST solution, where pendulum not vibrates but stay in lower position.

Similar analysis is performed for system with the activated MR damping about value of \( \alpha_3 = 0.15 \), Fig.6b. The introduced MRD causes the reduction of stable solution (rotation zones) and decrease of angular pendulum’s velocity. However, new very small rotation regions, located nearly \( \vartheta \approx 0.69 \) is observed. The change of stability solution occurs in the NS or BP bifurcation points.

Figure 7. Bifurcation diagrams \( \vartheta \) versus maximal angular pendulum’s velocity for \( \theta = 2, \beta_1 = 0.1, \beta_2 = 0.025, \alpha_3 = 0 \) (a) and basins of attraction of Fig.6a, for \( \vartheta = 2 \).

Figure 8. The average recovered power for system with SMA spring: \( \theta = 1.5, \vartheta = 1.5 \) (a), and \( \theta = 1.5, \vartheta = 0.7 \), during rotation motion for system with activated SMA spring, \( \Theta = 2, \beta_1 = 0.1, \beta_2 = 0.025 \).
The activation of SMA spring causes change in the stability and rotation zones expanded (Fig. 7a). Interestingly, that two possible stable rotations (with similar angular velocity) are possible. The rotation region increases to $\vartheta \approx 0.57 - 1.75$ and additionally the positive and negative rotations breaking symmetry.

Finally, the time histories of averaged power for rotational motion of the pendulum are presented. Figure 8a presents a recovered power during negative rotation, for point labelled as no.1 in Fig.7a. While, Fig.8b shows recovered averaged power for point no.2 (positive rotation). From the EH point of view, the rotations presented in Fig.7a is better, because of angular velocity is higher (module of P). Note, that maximal amplitude $\dot{\varphi}_{\text{max}}$ shows positive or negative values (no absolute).

4 CONCLUSIONS

This paper presents the pendulum absorber-harvester system with active suspension. The proposed conception concerned the application of energy harvesting device, mounted in/on the pivot of the pendulum absorber. The active suspension can be applied to control the dynamics. The suspension is dedicated especially for systems where control is difficult (systems with pendulum).

Introducing MRD or SMA spring practically does not decrease absorption effect (for analyzed parameters). Additionally, MRD slightly reduced rotation region and can introduce a new (Fig.6b). Moreover, MRD reduces the resonance region from both sides. The angular velocity is reduced, and recovered energy level decreased.

Introduction of SMA spring can significantly change dynamics. It is possible to increase (or decrease) of the pendulum’s velocity. Furthermore a new stable or unstable solutions and new bifurcation can exist. Especially the lower temperatures are dangerous (the martensite phases).

Unfortunately in these regions achieved angular velocities are highest.

The obtained results shows that this harvesto-absorber system can recover power on 30 – 40 milliwatts. Based on the investigation results, can be concluded that the system may works as dynamical damper and energy harvester simultaneously without loss its efficiency. In future, the experimental verification will be done on special laboratory rig.

Acknowledgments This work is financially supported under the project of National Science Centre according to decision no. DEC-2013/11/D/ST8/03311.

REFERENCES


ABSTRACT
Static balancing is often used in industrial applications in order to reduce the actuator efforts required during the machine operations. Literature offers a number of methods to study static balancing and to define positions and features of the balancing devices that are commonly used to achieve gravity compensation, i.e. counterweights, springs, hydro/pneumatic cylinders, and/or auxiliary parallelograms. This work deals with the static balancing of a commercial spatial manipulator, primarily conceived for palletizing tasks with a maximum payload of 150 kg. Since the robot has a hybrid architecture that includes closed kinematic chains, common balancing methods for serial manipulators are not suitable and new solutions must be developed. The paper proposes a number of different balancing solutions which include counterweights and springs, whose characteristics and positions are determined by means of an optimization algorithm. The effectiveness of each solution is investigated and the corresponding feasibility is discussed.

Keyword: static balancing, balanced manipulator, gravity compensation, optimization.

1 INTRODUCTION
This paper reports the study about the static balancing of a commercial spatial manipulator which has the primary task to palletize medium/high size items (maximum payload is 150 kg) working in a cylindrical workspace. The robot actuators are subjected to some criticalities in terms of overload and power consumption when performing certain trajectories. Instead of choosing to select more powerful actuators, the possibility to statically balance the manipulator (which presents heavy members) was targeted as a viable solution of the problem. Thanks to static balancing, actuators are indeed not required to sustain the weight of the moving parts. In this application, gravity loads are comparable to inertia loads since the robot operates at low/moderate dynamics. Therefore the removal of the gravitational contribution from the actuator load should solve the overload problems [1, 2]. It may be even considered to possibly replace current actuators with lower size ones, due to the expected drop of their required performance.

As known, static balancing of a mechanism is achieved if its total potential energy is made invariant for any admissible pose. This could be done with a proper distribution of the link inertias and/or introducing balancing elements such as counterweights, springs, hydro/pneumatic cylinders, and auxiliary linkages. Different approaches and solutions for static balancing of mechanisms can be found in the literature, e.g. [3, 4]. Even though the robot is a serial manipulator from the functional viewpoint, it should be noted that its architecture presents closed-loop chains featuring linear actuators. Hence, balancing techniques commonly adopted for serial manipulators are not directly applicable. As a consequence, the implementation of a specific solution for gravity compensation is required.

In this study a number of balancing solutions with counterweights and/or springs are proposed and analysed independently. The features of the balancing elements (such as the value and location of the additional masses, the spring stiffness constants and dimensions) are chosen through two different optimization algorithms. One algorithm is conceived when at least one
counterweight is introduced and is based on the minimization of a constrained nonlinear multivariable function. The other one solves a multi-objective constrained nonlinear multivariable problem, when the only balancing elements are springs. In order to (partly) evaluate the effectiveness and feasibility of the proposed solutions, dynamic analyses were carried out in order to estimate the global efforts required to the actuators (including inertia loads) when executing critical trajectories. Since the main purpose of the study focuses on the robot energy efficiency, the RMS values of the motor loads are used as a basic metrics to estimate the robot power consumption for dynamic operations.

2 MATERIALS AND METHODS

2.1 Description of manipulator

The studied palletizing robot has 4 degrees of freedom (DOFs) to position and orient with respect to the vertical axis the pallets that it manipulates. From a functional viewpoint it can be classified as a cylindrical robot, even though its architecture does not present the succession of three revolute joints typical of the serial cylindrical manipulator. Figure 1 illustrates the schematic kinematic chain of the robot.

*Actuator 1* provides rotational motion around a fixed vertical axis of all the moving links. A planar 2-DOF closed kinematic chain, driven by two actuated prismatic joints (*Actuator 2* and *Actuator 3*), permits the translation of the end-effector in a vertical plane: the linkage works like a pantograph and motion is decoupled, i.e. *Actuator 2* drives the motion along the horizontal direction only (with a scale factor of +6) whereas *Actuator 3* controls the vertical motion only (motion scaled by factor -5). The kinematic scheme of the mechanism, with links arranged to form three parallelograms, allows the axis of the palletizing gripper to remain vertical in all possible configurations of the manipulator inside its workspace. *Actuator 4* controls the gripper orientation around this vertical axis.

The ranges of motion of the four actuated joints are shown in Table 1. For the results of the dynamic analyses presented in following sections, it is worth highlighting that the typical control of the robot foresees the implementation of joint motion laws with trapezoidal acceleration and that, in front of a maximum payload of 150 kg, the robot overall mass is about 2000 kg.

![Schematic of the palletizing manipulator.](image)

**Figure 1.** Schematic of the palletizing manipulator.

<table>
<thead>
<tr>
<th>Actuator 1</th>
<th>Actuator 2</th>
<th>Actuator 3</th>
<th>Actuator 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0°-360°</td>
<td>0 - 266 mm</td>
<td>0 - 472 mm</td>
<td>0°-360°</td>
</tr>
</tbody>
</table>
### 2.2 Static balancing

Static balancing of a mechanism is achieved if its total potential energy is made invariant for any admissible pose. Gravity compensation can be performed by adding balancing devices that make nought the partial derivatives of the system total potential energy $U$ with respect to all the mechanism generalized coordinates $q_k$:

\[
\frac{\partial U}{\partial q_k} = 0
\]  

(1)

The effects of the required balancing devices on the mechanism dynamic operation are evaluated through a numerical model based on the Lagrange equations of motion. In particular, the generalized forces $Q_k$ (which in this case coincide with the motor actions) are computed by inverse dynamic analysis:

\[
Q_k = \frac{d}{dt} \left( \frac{\partial E}{\partial \dot{q}_k} \right) - \frac{\partial E}{\partial q_k} + \frac{\partial U}{\partial q_k}
\]  

(2)

where $E$ is kinetic energy.

Actuators 1 and 4 provide rotation around vertical axes, and therefore they do not have to sustain any gravity load. However the required motor action of Actuator 1 may be significantly affected by the introduction of balancing devices attached to the robot moving parts. Hence Actuator 1 is taken into account for static balancing optimization. Actuator 4 is neglected since it is only affected by the payload carried by the gripper. Therefore the dynamic model is described by the three generalized coordinates $\alpha, \beta$ and $\gamma$ (Fig. 1).

This work studies balancing devices based on counterweights and/or springs, adopted alternatively or simultaneously. Attaching counterweights to the mechanism modifies Equation (2) into:

\[
Q_k = \frac{d}{dt} \left( \frac{\partial E}{\partial \dot{q}_k} \right) + \frac{d}{dt} \left( \frac{\partial E_{mass}}{\partial \dot{q}_k} \right) - \frac{\partial E}{\partial q_k} - \frac{\partial E_{mass}}{\partial q_k} + \frac{\partial U}{\partial q_k} + \frac{\partial U_{mass}}{\partial q_k}
\]  

(3)

where $E_{mass}$ and $U_{mass}$ are the total kinetic energy and total potential energy contribution given by the balancing counterweights, respectively. Gravity compensation is achieved by imposing:

\[
\frac{\partial U}{\partial q_k} + \frac{\partial U_{mass}}{\partial q_k} = 0
\]  

(4)

Therefore, the dynamics of the balanced robot is described by:

\[
Q_k = \frac{d}{dt} \left( \frac{\partial E}{\partial \dot{q}_k} \right) + \frac{d}{dt} \left( \frac{\partial E_{mass}}{\partial \dot{q}_k} \right) - \frac{\partial E}{\partial q_k} - \frac{\partial E_{mass}}{\partial q_k}
\]  

(5)

The main objective is to determine optimal parameters of the $i$-th counterweight, namely its mass, $m_i$, and its location (described by a proper scalar, $b_i$), whereas, at this stage of the research, optimizing the counterweight shape is not considered essential. Hence an arbitrary geometry is assumed for estimating the inertia tensor of each counterweight. The $i$-th balancing counterweight is considered as a simple lead disk having radius, $r_i$, four times bigger than height, $h_i$. Thanks to such hypothesis on the geometry, the height can be expressed as a function of the counterweight mass

\[
h_i = \sqrt[3]{\frac{m_i}{16\rho}} \quad ; \quad r_i = 4h_i
\]  

(6)

and consequently the inertia tensor of each counterweight, referred to the local principal axes,
is defined by the counterweight mass \( m_i \).

As for the introduction of elastic elements, the mass and inertia of balancing springs required for achieving gravity compensation is normally rather negligible with respect to those of the manipulator moving parts. Hence, at this stage of the research, balancing springs are considered as massless. Under this assumption, the introduction of balancing springs modifies Equation (2) as follows:

\[
\begin{bmatrix}
\frac{1}{12} m_i (3 r_i^2 + h_i^2) & 0 & 0 \\
0 & \frac{1}{12} m_i (3 r_i^2 + h_i^2) & 0 \\
0 & 0 & \frac{1}{2} m_i r_i^2
\end{bmatrix}
\]

(7)

where \( U_{spring} \) is the total potential energy stored in springs. Gravity compensation is achieved by imposing

\[
\frac{\partial U}{\partial q_k} + \frac{\partial U_{spring}}{\partial q_k} = 0
\]

(9)

hence obtaining

\[
Q_k = \frac{d}{dt} \left( \frac{\partial E}{\partial q_k} \right) - \frac{\partial E}{\partial q_k}
\]

(10)

Therefore, due to the adopted simplification, generalized forces are not affected by balancing springs and all the robot variants balanced by using only springs result equivalent from the dynamic point of view.

For the \( j \)-th spring, the spring rate, \( K_j \), and the two spring anchor points (described by scalars \( p_j \) and \( l_j \)) must be computed. Further design parameters of both tension and compression springs are considered in order to better evaluate the feasibility of the proposed solutions, by performing comparisons with the specifications of real springs. In particular, the spring mean diameter (\( D_j \)), the wire diameter (\( d_j \)) and the number of coils (\( n \)) are taken into account in the optimization process.

2.3 Static balancing optimization

This works primarily aims at reducing the robot power consumption for typical functioning conditions. However Equation (10) proves that inverse dynamics analysis can not provide suitable criteria for optimizing balancing springs, under the considered assumption of massless springs. Therefore two different optimization strategies are implemented: the first one can be adopted when only springs are installed, whereas the second one is valid when at least one counterweight is introduced.

2.3.1 Balancing solutions with only springs

Each balancing spring is optimized by minimizing two distinct functions, namely the spring maximum force exerted throughout the workspace (\( Spring_{max} \)) and the spring mass (\( Spring_{mass} \)). The former is adopted to limit the increment in the reaction forces of some robot joints, which is commonly associated with the introduction of balancing springs. The latter is used to limit both the dimensions and the cost of the spring. Hence, even if the spring inertia properties are neglected for the inverse dynamics problem, minimization of the spring mass is taken into account to increase the feasibility of the balancing solution. The use of such objective functions results in a multi-objective constrained nonlinear multi-variable problem [5]. The following vector of unknowns
\[ x_j = (K_j, p_j, l_j, d_j, D_j, n_j) \]  

where \( j = 1, \ldots, S \) (\( S \) being the total number of springs adopted in the specific balancing solution), is computed for each spring by using the method described in [6] to minimize the following system of objective functions:

\[
\begin{align*}
\text{Spring}_{Fj} &= K_j(L_{\max,j} - L_{\min,j}) \\
\text{Spring}_{Mj} &= \frac{\rho \pi^2 d_j^2 D_j (n_j + 2)}{4}
\end{align*}
\]  

(12)

where \( L_{\min,j} \) and \( L_{\max,j} \) are, respectively, the minimum and the maximum length reached by the spring throughout the workspace (which depend on \( p_j, l_j \) and on the robot geometry). In addition to the static balancing condition expressed by Equation (9), for each spring the following design constraints must be satisfied (see [7, 8]):

\[
\begin{align*}
K_j &= \frac{Gd_j^4}{8D_j^3 n_j} \\
C_{\text{min}} &\leq C_j \leq C_{\text{max}} \\
L_0 &\leq L_{\min} \quad \text{(tension spring only)} \quad \tau_{\max} \leq \tau_a \\
\frac{L_0}{D_j} &\leq B_j \quad \text{(compression spring only)} \\
\frac{\text{pitch}}{\pi D_j} &\leq \tan \theta \quad \text{(compression spring only)} \\
x_{\text{min}} &\leq x_j \leq x_{\text{max}}
\end{align*}
\]  

(13)

where \( G \) is the shear modulus of the spring material, \( L_0 \) is the spring free length, \( B_j \) is a parameter determined by spring buckling criteria, \( \theta \) is the critical pitch angle (normally about 12 deg), \( C_j \) is the spring index (i.e. \( D_j/d_j \)), \( \tau_a \) is maximum allowable shear stress of the material, and \( \tau_{\max} \) is the maximum shear stress (given the Wahl factor, \( K' \)):

\[ \tau_{\max} = \frac{8K'Spring_{Fj}D}{\pi d^3} \]  

(14)

2.3.2 Balancing solutions with counterweights

An analysis algorithm, referred to as search algorithm hereafter, initially solves the inverse dynamics problem of the unbalanced robot to find the most critical joint trajectories for the motors. In particular the algorithm finds, for each actuator, the trajectory that generates the highest RMS value and the one characterized by the highest peak value of the required motor action. The trajectories with critical RMS are used for defining the objective function, while the others are taken into account as constraint functions. It is worth noting that the critical joint trajectories are not necessarily the same for each actuator, i.e. they could correspond to different paths of the end-effector in the Cartesian Space.

A single objective function is adopted for optimizing the counterweights. At the first iteration step, the function is defined as the power mean of the terms \( RMS_r \), each one providing the RMS of the driving action of the \( r \)-th actuator along the corresponding critical trajectory \((r = 1, 2, 3)\). Indeed in preliminary analyses the power mean was found more effective than the other tested parameters (e.g. the total RMS or the simple mean of the critical RMS) for achieving a reduction of the required motor action. It may be noticed that while the RMS values computed by the search algorithm for the unbalanced robot are scalars, each \( RMS \) parameter included in the objective function is an expression that depends on the unknown balancing variables. A constrained nonlinear multivariable problem [9] is therefore defined. A solving algorithm based on the interior-point method [10] is adopted for finding the unknown parameters.
where \( N \) is the number of counterweight to be installed, by minimizing the function

\[
RMS^*(x) = \sqrt[3]{\frac{1}{3} (RMS^2_1 + RMS^2_2 + RMS^2_3)}
\]

(16)

In addition to the balancing condition provided by Equation (4), the following constraint functions must be satisfied:

\[
\left\{ \begin{array}{l}
PEAK_r \leq PEAK_{max,r} \\
x_{\min} \leq x \leq x_{\max}
\end{array} \right.
\]

(17)

where, \( PEAK_{max} \) is the maximum allowed overload and subscript \( r \) refers to the \( r-th \) actuator.

After the first optimization step, the search algorithm is applied to the balanced mechanism. If critical trajectories different from the ones taken into account at the previous iteration are found, they are included in the objective function. For example, if a new critical trajectory is found for Actuator 2, we insert \( RMS_{2new} \) in the power mean expression and Equation (16) becomes:

\[
RMS(x)^* = \sqrt[3]{\frac{1}{3} (RMS^2_1 + RMS^2_2 + RMS^2_3 + RMS^2_{2new})}
\]

(18)

New critical PEAK values are included in the constraints as well. The search algorithm and the optimization process are iteratively applied to the balanced mechanism until new critical trajectories are no more identified.

In case the static balancing solution includes springs as well, the spring parameters required for achieving balancing are added to the vector of the unknown variables to be optimized:

\[
x = (m_1, \ldots, m_N, b_1, \ldots, b_N, K_s, \ldots, K_S, p_1, \ldots, p_S, l_1, \ldots, l_S)
\]

(19)

where \( N \) is the number of counterweights and \( S \) is the number of springs in the balancing solution. Then the formulation described by Equations (16-17) is adopted for optimization purpose.

### 3 PROPOSED BALANCING SOLUTIONS

Four interesting balancing solutions, among all the studied ones, are here illustrated in details and discussed. The search algorithm described in Section 2.3.2 is applied to the unbalanced robot and to all the examined balanced variants (after optimization) in order to evaluate the improvement achieved through gravity compensation. The critical RMS and peak values of the motor loads computed for the unbalanced manipulator with unloaded gripper are reported in Table 2.

**Table 2. Critical RMS and Peak motor loads of the unbalanced robot.**

<table>
<thead>
<tr>
<th>Actuator</th>
<th>RMS [Nm]</th>
<th>PEAK [Nm]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actuator 1</td>
<td>35.0</td>
<td>48.0</td>
</tr>
<tr>
<td>Actuator 2</td>
<td>16.6</td>
<td>22.7</td>
</tr>
<tr>
<td>Actuator 3</td>
<td>22.6</td>
<td>32.5</td>
</tr>
</tbody>
</table>

### 3.1 Solution 1

This solution requires the introduction of two counterweight (Fig. 2). The variables of the optimization algorithm are:
After two iterations of the optimization process, the balancing parameters reported in Table 3 are obtained. Minus sign for the $b_i$ parameter says that the counterweight should be located on the opposite side with respect to that shown in Fig. 2.

**Table 3.** Optimal balancing parameters of Solution 1.

<table>
<thead>
<tr>
<th>Counterweight</th>
<th>$m_i$ [Kg]</th>
<th>$b_i$ [mm]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Counterweight 1</td>
<td>228.6</td>
<td>-38.0</td>
</tr>
<tr>
<td>Counterweight 2</td>
<td>678.8</td>
<td>353.7</td>
</tr>
</tbody>
</table>

Table 4 shows the results concerning the dynamic effects of balancing on the actuators provided by the search algorithm.

**Table 4.** Critical RMS and Peak motor loads for Solution 1.

<table>
<thead>
<tr>
<th>Actuator</th>
<th>RMS [Nm]</th>
<th>$\Delta$RMS$%$</th>
<th>PEAK [Nm]</th>
<th>$\Delta$PEAK$%$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actuator 1</td>
<td>36.4</td>
<td>+4</td>
<td>50.0</td>
<td>+4.2</td>
</tr>
<tr>
<td>Actuator 2</td>
<td>9.1</td>
<td>-45.2</td>
<td>14.2</td>
<td>-37.4</td>
</tr>
<tr>
<td>Actuator 3</td>
<td>8.9</td>
<td>-60.6</td>
<td>18.5</td>
<td>-43.1</td>
</tr>
</tbody>
</table>

Gravity compensation provides a satisfactory enhancement of the robot operation in terms of reduced motor loads. However, the balancing counterweights remarkably increase the robot mass, from about 2000 Kg to about 2900 Kg (+45%). In addition, this solution appears hardly achievable in practice since bulky counterweights should be attached to the robot and interference with the robot links could be an issue. Therefore further investigations on the solution feasibility seem advisable.

### 3.2 Solution 2

This solution features two springs (Fig. 3). The variables of the optimization algorithm are

$$x = (m_1, b_1, m_2, b_2)$$

(21)
The parameters provided by the optimization algorithms are reported in Table 5. Minus sign for the spring ratio is associated with the use of a compression spring. The RMS and the peak values of the critical motor loads computed for this balanced variant are shown in Table 6.

### Table 5. Optimal balancing parameters of Solution 2.

<table>
<thead>
<tr>
<th></th>
<th>$K_j$ [N/mm]</th>
<th>$l_j$ [mm]</th>
<th>$q_j$ [mm]</th>
<th>$F_{max}$ [N]</th>
<th>Mass [Kg]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring 1</td>
<td>-3.738</td>
<td>300</td>
<td>600</td>
<td>1522</td>
<td>22</td>
</tr>
<tr>
<td>Spring 2</td>
<td>11.794</td>
<td>300</td>
<td>800</td>
<td>4500</td>
<td>15</td>
</tr>
</tbody>
</table>

### Table 6. Critical RMS and Peak motor loads for Solution 2.

<table>
<thead>
<tr>
<th></th>
<th>RMS [Nm]</th>
<th>ΔRMS$%$</th>
<th>PEAK [Nm]</th>
<th>ΔPEAK$%$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actuator 1</td>
<td>35.0</td>
<td>0</td>
<td>48.0</td>
<td>0</td>
</tr>
<tr>
<td>Actuator 2</td>
<td>9.2</td>
<td>-44.6</td>
<td>14.3</td>
<td>-37.0</td>
</tr>
<tr>
<td>Actuator 3</td>
<td>6.2</td>
<td>-72.6</td>
<td>12.8</td>
<td>-60.6</td>
</tr>
</tbody>
</table>

The increment in the manipulator mass is rather limited: from 2000 Kg to 2037 Kg (about +2%). This confirm the soundness of the initial assumption, i.e. that the installation of balancing springs is expected to marginally affect the motor loads required for dynamic operation.

The characteristics of the required balancing elements are comparable to the specifications of common tension and compression springs. Hence this solution appears feasible to be implemented in practice.

### 3.3 Solution 3

This solution requires the installation of three springs, as shown in Fig. 4. The variables of the optimization algorithm are

$$x = (K_1, K_2, K_3, p_1, l_1, l_2, d_1, d_2, d_3, D_1, D_2, D_3, n_1, n_2, n_3) \quad (22)$$

The results of optimization are shown in Table 6. Since this solution is dynamically equivalent to Solution 2, the results concerning the dynamic effects of balancing are the same as well (Table 6).

The same conclusions discussed for Solution 2 hold. Furthermore, even if an additional spring is required, locating Spring 1 and 2 on the back of the rotating turret limits possible interference.
with members during motion. Hence this solution may result particularly convenient for possible practical implementation.

![Schematic of balancing Solution 3.](image)

**Figure 4.** Schematic of balancing Solution 3.

<table>
<thead>
<tr>
<th></th>
<th>$K_j$ [N/mm]</th>
<th>$l_j$ [mm]</th>
<th>$p_j$ [mm]</th>
<th>$F_{\text{max}}$ [N]</th>
<th>Mass [Kg]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring 1</td>
<td>7.4745</td>
<td>300</td>
<td>440.91</td>
<td>3528</td>
<td>4.6</td>
</tr>
<tr>
<td>Spring 2</td>
<td>-7.4745</td>
<td>-</td>
<td>440.91</td>
<td>3528</td>
<td>4.4</td>
</tr>
<tr>
<td>Spring 3</td>
<td>9.4588</td>
<td>300</td>
<td>800</td>
<td>3608</td>
<td>7.4</td>
</tr>
</tbody>
</table>

### 3.4 Solution 4

This solution requires one counterweight and one springs (Fig. 5). The variables of optimization algorithm are

$$\mathbf{x} = (m, b, K, p, l)$$  \hspace{1cm} (23)

![Schematic of balancing Solution 4.](image)

**Figure 5.** Schematic of balancing Solution 4.
The results after two optimization steps are reported in Table 8.

<table>
<thead>
<tr>
<th>Counterweight</th>
<th>( m ) [Kg]</th>
<th>( b ) [mm]</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>228.6</td>
<td>-176.8</td>
</tr>
</tbody>
</table>

The mass of the manipulator rises to about 2250 Kg (+13%). This mass increment is remarkably smaller than for Solution 1. Even if a counterweight is adopted, the dynamic results are comparable to those provided by solutions involving only springs. Since attaching counterweights is generally easier than installing springs, also this solution appears particularly promising.

### 3.5 Discussion

In order to quantify the effectiveness of the different solutions from the functional viewpoint, the actuator driving actions are compared in terms of RMS and peak values. In particular the percentile variations with respect to the unbalanced manipulator are computed as:

\[
\Delta \text{RMS}\% = \left( \frac{\text{RMS}_B - \text{RMS}_U}{\text{RMS}_U} \right) \times 100
\]

\[
\Delta \text{PEAK}\% = \left( \frac{\text{PEAK}_B - \text{PEAK}_U}{\text{PEAK}_U} \right) \times 100
\]

where the subscripts B and U refer to balanced and unbalanced manipulator, respectively.

Results are reported in Tables 2, 4, 6, and 9, and show that all the proposed solutions satisfy the main goal of the study, that is to decrease the actuator efforts required during motion, in particular for Actuators 2 and 3. Solution 1 appears worse than the other three, which are basically equivalent. The reason of this difference can be found in the initial hypothesis of zero-mass of springs in the definition of the dynamic model (Lagrange equations) of the manipulator.

It is worth noticing that the “hybrid” Solution 4, which presents one balancing mass and one balancing spring, features the same improvement of Solutions 2 and 3, with only balancing springs. All solutions increase the overall mass of robot, in particular solutions 1 and 4 which foresees the introduction of counterweights.

### 4 CONCLUSION

This paper presents the study relative to the static balancing of a palletizing industrial manipulator, characterized by an architecture featuring closed-loop chains devised to drive the motion in a vertical plane of the end-effector carrying heavy loads. The elimination of gravitational load contributions from the actuator actions permits to solve problems of actuator overloads and power consumption. A number of balanced variants of the manipulator were studied, introducing counterweights and/or springs to the members of the original (unbalanced) version. The balancing parameters (e.g. entity of additional masses, spring stiffness constants etc.) were determined by means of an optimization procedure which is described in the paper. Two different optimization algorithms were proposed, depending on the presence/absence of additional masses among the balancing elements. Four solutions are illustrated and discussed:
one solution presents two counterweights, two solutions present two and three springs, respectively, and one solution presents one counterweight and one spring. The solution with three springs appears as the most promising, both considering functional aspects and practical implementation. The next step of the activity is to complete the feasibility study of the targeted variant of the manipulator by defining in details the design modifications to introduce in the current manipulator architecture.

REFERENCES


ABSTRACT

Flexible multibody dynamics (FMD) has found many applications in control, analysis and design of mechanical systems. FMD together with the theory of structural optimization can be used for designing multibody systems with bodies which are lighter, but stronger. Topology optimization of static structures is an active research topic in structural mechanics. However, the extension to the dynamic case is less investigated as one has to face serious numerical difficulties. One way of extending static structural topology optimization to topology optimization of dynamic flexible multibody system with large rotational and transitional motion is investigated in this paper. The optimization can be performed simultaneously on all flexible bodies. The simulation part of optimization is based on an FEM approach together with modal reduction. The resulting nonlinear differential-algebraic systems are solved with the error controlled integrator IDA (Sundials) wrapped into Python environment by Assimulo [1]. A modified formulation of solid isometric material with penalization (SIMP) method is suggested to avoid numerical instabilities and convergence failures of the optimizer. Sensitivity analysis is central in structural optimization. The sensitivities are approximated to circumvent the expensive calculations. The provided examples show that the method is indeed suitable for optimizing a wide range of multibody systems. Standard SIMP method in structural topology optimization suggests stiffness penalization. To overcome the problem of instabilities and mesh distortion in the dynamic case we consider here additionally element mass penalization.

Keywords: Structural topology optimization, flexible multibody dynamics, transient response, SIMP.

1 INTRODUCTION

When designing a mechanical system where the main task is carrying or transferring a load, an economical product is obtained if the same functionality can be achieved by using a lighter structure. It can be accomplished by minimizing or maximizing an objective function which represents the quality of the system in the limits of given constraints. Structural optimization is about finding the best design where the main task is carrying a load. In particular, topology optimization (TO), a branch of structural optimization, is a part of conceptual design of a product. In TO, optimization starts from an initial model which is mostly a box called design space. A design is characterized by the material distribution in the design space. The design space is discretized by finite elements. Each finite element represents a design variable. The design variable ranges between a given upper bound exhibiting the material state and a lower bound exhibiting a hole. The design variable is selected to be the normalized density or the normalized thickness in case of a planar structure. The optimization algorithm iteratively changes the design variable to reach an optimized hole-material state. In many TO problems the goal is to minimize the deformation of the structure as a response to a prescribed load. In this case the objective function can be defined as the strain energy stored in the structure (compliance) which must be minimized within the limiting constraints – the total amount of the material [2, 3].
Topology optimization of static structures was subject of extensive research on methods. However, the extension to the dynamic case is less investigated. One strategy for topology optimization on a single body under transient loads is called component-based approach, [10]. In the component-based approach multiple static loads are selected from the transient loads acting on the isolated body. Thus, it is assumed that there is enough time for the body to settle before the load changes. This assumption is not realistic in case when the bodies encounter high accelerations. On the other hand, the shape and the weight of the body change in every optimization step, if the transient loads depend on the design. For instance, in a multibody system (MBS), the dynamic behavior and forces at joints change accordingly, hence the selected load cases are not valid anymore, cf. [14].

Another strategy for dynamic response structural optimization is the equivalent static loads method (ESLM), [8, 6]. In this approach the body is isolated from the rest of the system where all forces including the inertia forces are accounted for. A set of equivalent quasi-static load cases in every time step must be defined which produces the same displacement field as the one caused by dynamic loads. Then it would be possible to use the theory of the static structural optimization directly. However, ESLM is mostly developed for size and shape optimization. Using this method for topology optimization causes instability and failure of the optimization algorithm. In [6] this problem is attenuated by removing some of elements and updating the grid data in every optimization process. This approach has to restrict the design area and later revival of removed elements cannot be treated. Moreover, the element removal needs post processing of the data which is not unique for different problems. In addition, since the body is isolated from the rest of the system, constraints and the objective function cannot be defined based on the overall system response, [14].

We present here an alternative approach treating topology optimization of all flexible bodies simultaneously while they are operating in an MBS based on the system overall response considering all transient reaction and inertia forces. In this paper this approach is called topology optimization of a multibody system (TOMBS). In [5] a related approach is used with two different regimes of stiffness penalization. The switching criteria between two regimes might differ between problems, so that this formulation is not always applicable. Here it is argued that the origin of the numerical difficulties and mesh distortion which result in non-convergence of the optimization algorithm suggested by SIMP, [2], is an effect of what we call flying elements. To reduce this effect we suggest element mass penalization in addition to stiffness penalization.

The suggested method in dynamic response topology optimization is demonstrated by two simple two-dimensional MBS. Sensitivity expressions are approximated by eliminating terms which are assumed to have low order of magnitude but are numerically expensive to calculate. Achieving convergence of the optimization algorithm in a reasonable computation time in problems with large number of design variables proves that the above assumption is valid for a wide range of multibody systems, though we also observed convergence problems in case when the body experiences acceleration peaks. The approach is applicable for designing vehicle components, high-speed robotic manipulators, air planes and space structures.

2 Optimization Problem

The optimization problem for a flexible multibody system is mathematically described as
Find \( X^j \in \mathbb{R}^n \)

which minimizes \( \int_0^{t_s} C^j(X^j, q^j, t) \, dt \) (1)

subject to

\[
\begin{align*}
&M'(X^i, q^i)\ddot{q}^i + K'(X^i)q^i = f'(X^i, q^i) \quad i = 1, 2, \ldots, n_b \\
&C(q, t) = 0
\end{align*}
\]

(2)

and

\[
\begin{align*}
&g_1(X^j) = \int_{A^j} X^j \, da - V_{\text{max}}^j \leq 0 \\
&X_{j, \text{min}} \leq X^j \leq X_{j, \text{max}}
\end{align*}
\]

(3)

where, \( j \) is the index of the body which will be optimized, \( C \) denotes the compliance, \( X \) is the vector of design variables, \( q^j \) is the vector of elastic coordinates or displacement vector, \( t \) is running time, and \( t_s \) is the simulation stopping time. Eq. (2) states the equality constraint which is a system of nonlinear differential algebraic equations describing the motion of the flexible multibody system. In that equation \( i \) is the index of the bodies in the system, \( q \) is the total vector of rigid and elastic coordinates; in a planar body the first three elements of the vector \( q \) denote the rigid body coordinates, the global position of the body reference point and its orientation; the other elements of the vector \( q \) are the same as \( q^j \). \( M \) is the mass matrix, \( \ddot{q} \) is the total vector of rigid and elastic accelerations, \( K \) is the body stiffness matrix associated with rigid and elastic coordinates, \( f \) is the force vector which accounts for reaction forces, external forces and Coriolis and centrifugal terms, \( C(q, t) \) is the vector of kinematic algebraic constraint equations describing the MBS joints and prescribed trajectories, \( n_b \) is the total number of bodies in MBS.

In Eqs. (3) \( g_1 \) denotes the inequality constraints, \( A \) is the total area of the body and \( V_{\text{max}} \) is the maximum allowable normalized volume, \( X_{\text{min}} \) and \( X_{\text{max}} \) are the lower and upper bounds of the design variable also called a box constraint.

According to the floating frame of reference approach, the mass matrix is composed by submatrices corresponding to rigid coordinates, elastic coordinates and coupling terms; in planar mechanism, three first rows and columns of the body stiffness matrix are entirely zero. The nonzero submatrix, \( K_{ff} \), corresponds to the elastic coordinates which are a function of the design variable. A detailed derivation of the mass matrix and the stiffness matrix can be found in [11, 4].

We consider a non-weighted time integration of the compliance as the objective function for notational simplicity. In practice a weight function has to be introduced in Eq. (1) to give small but relevant peaks in the objective function a higher influence in the minimization process, [9, 6].

The relation between the design variables \( X \) and the displacement vector \( q \) is given by equality constraints Eq. (2). Having solved the dynamic equation of motion and replacing the integration by a summation the objective function has the following form when scaled by \( t_s/s \)

\[
f^j = \sum_{l=0}^{s} C^j_l(X^j)
\]

where \( C^j_l \) is the compliance at time step \( l \) and \( s \) is the total number of time steps. The compliance is given by

\[
C^j_l(X^j) = q^j_l \, ^T(X^j)K^j_{ff}q^j_l(X^j),
\]

(4)

where, \( q^j_l \) is the elastic displacement vector of body \( j \) at time step \( l \), \( K^j_{ff} \) is the stiffness matrix associated to the elastic coordinates which is obtained by finite element analysis.

The solid isotropic material with penalization approach (SIMP) is widely used for topology optimization problems. SIMP is based on the convex linearization method (CONLIN) or the optimality
criteria (OC) method which are gradient based methods, [2, 3]. The big advantage of CONLIN and OC methods is that they make an explicit and convex approximation of the objective and constraint functions. More importantly the result of the approximation is a separable function with respect to the design variable. These properties make it possible to find a local minimum in an efficient way when the number of design variables is large.

The idea of both CONLIN and OC is to linearize the objective and constraint functions at the intermediate variable $Y(X^k)$ by writing the two first terms of their Taylor expansion at $Y(X^k)$, where, $X^k$ is the design variable at iteration $k$ which is a constant vector; then, to solve the optimization sub-problem in the vicinity of $X^k$ with Lagrangian duality method. The solution of the subproblem, $X^{(k+1)}$, is then assigned to $X^k$ and the method is repeated until convergence is achieved. The intermediate variable is chosen such that the objective or constraint functions become more close to a linear function; thus the linearization at iteration step $k$ introduces a smaller approximation error. The convergence criterion can be the change of the objective function from one iteration to the next or the change in the norm of the design variable, $||X^{(k+1)} - X^k|| < \varepsilon$, where, $\varepsilon$ is a given threshold. Filters are also important in topology optimization [12]. The same filters as in the static response topology optimization are applicable in the dynamic case also. In particular, the mean sensitivity filter is used for the examples in a later section.

In Figure 1 the general steps of TOMBS are illustrated.

![Figure 1. General steps of TOMBS.](image)

### 2.1 Solving the Differential Algebraic Equation of Motion

An independent code was developed by the authors for building and simulating a planar flexible multibody system with the purpose of implementing and conceptional testing TOMBS. The simulation is based on a floating frame of reference approach and finite element formalism together with modal reduction and static condensation, Craig-Bampton method [11, 13]. The resulting nonlinear differential-algebraic system is solved with the error controlled integrator IDA (Sundials) wrapped into a Python environment by Assimulo [1]. The code is also interfaced to Dymola to allow its verification.

Due to the large number of degrees of freedom occurring in topology optimization problems, the simulation would not be possible without the use of a model reducing method, e.g. modal reduction. Moreover, the simulation must be repeated with the updated thickness in every optimization iteration, thus it is important to reduce the simulation time. On the other hand, for modal reduction, an eigenvalue problem has to be solved and full coordinates must be retained for the entire system in every iteration step.
2.2 Sensitivity Approximations and Optimization Subproblem

CONLIN or OC approximation of the objective and constraint functions can be done as follows

\[ C'_j(X^j) = C'_j(X^{j,k}) + \sum_{e=1}^{n} \frac{\partial C'_j(X^{j,k})}{\partial X^{j,k}_e} \frac{\partial X^{j,k}_e}{\partial Y^j_l} (Y^j_l(X_e^j) - Y^j_l(X_e^{j,k})) \]  \hspace{1cm} (5)

Subscript \( e \) denotes the design variable or the finite element index. The choice of the intermediate variable \( Y(X^k) \) depends on the function to be linearized, \( C'_j(X^j) \). A good choice of \( Y(X^k) \) results in a fast convergence of the optimization algorithm.

In order to evaluate Eq. (5) it is necessary to compute the sensitivity of the objective function with respect to the design variable at iteration \( k \) at time \( t_l \). Calculating the sensitivity numerically by direct methods is very expensive. Here, we assume for some derivatives

\[ \frac{\partial f^j}{\partial X^j_i} \approx 0, \quad \frac{\partial M^j}{\partial X^j_i} \approx 0, \quad \frac{\partial q^j}{\partial X^j_i} \approx 0 \]  \hspace{1cm} (6)

holds and they can be eliminated from the sensitivity expression. This leads to a simplified computation of the sensitivity of the dynamic response problem which becomes comparable to that in the static response problem. Another alternative is to use adjoint method [9, 7], which is planned to be tested in future versions of TOMBS.

It was observed in practical computations, where the bodies in the system do not experience sudden change of position or high accelerations, that the above made approximations do not harm convergence of the optimization algorithm. However, it would not be possible without mass penalization introduced in Section 2.3. The sensitivity of the compliance can be calculated as follows

\[ \frac{\partial C'_j(X^j)}{\partial X^j_i} = \frac{\partial q^j_{f,j}^T(X^j)}{\partial X^j_i} K_{f,j} q^j_{f,j} (X^j) - q^j_{f,j} (X^j) \frac{\partial \mathbf{K}^j_{f,j}}{\partial X^j_i} q^j_{f,j} (X^j) \]  \hspace{1cm} (7)

\[ + q^j_{f,j} (X^j) K_{f,j} \frac{\partial q^j_{f,j} (X^j)}{\partial X^j_i} \]

\[ \frac{\partial q^j_{f,j} (X^j)}{\partial X^j_i} \] can be found by differentiating the equilibrium constraint with respect to the design variable. The differential part of the equation of motion is

\[ M^i \ddot{q}^i + \mathbf{K}^i \dot{q}^i = f^i, \quad i = 1, 2, \ldots, n_b \]  \hspace{1cm} (8)

Differentiating Eq. (8) with respect to \( X^j_i \) gives

\[ M^i \frac{\partial \ddot{q}^i}{\partial X^j_i} + K^i \frac{\partial \dot{q}^i}{\partial X^j_i} = \frac{\partial f^i}{\partial X^j_i} - \frac{\partial M^i}{\partial X^j_i} \dot{q}^i - \frac{\partial K^i}{\partial X^j_i} q^i \]  \hspace{1cm} (9)

Differential equation Eq. (9) together with the constraint equations \( \mathbf{c}(\mathbf{q}, t) = 0 \) has the same form as the equations of motion (2) which have to be solved numerically. However, Eq. (9) is needed to be solved for all times \( (s+1) \) and for every element of the body. For topology optimization the number of design variables or finite elements often is large. Large number of design variables and time steps make finding the sensitivity very expensive. To circumvent this problem we eliminate terms (6) and also assume that the sensitivity of the elastic coordinates of a body, for instance body \( j \), does not depend on other bodies, \( \frac{\partial \mathbf{K}^j_{f,j}}{\partial X^j_i} = 0 \) and \( \frac{\partial \mathbf{q}^j_{f,j}}{\partial X^j_i} = 0 \) if \( i \neq j \), then Eq. (9) is simplified to

\[ \frac{\partial \mathbf{K}^j_{f,j}}{\partial X^j_i} q^j_{f,j} + K^j_{f,j} \frac{\partial q^j_{f,j}}{\partial X^j_i} = 0 \]  \hspace{1cm} (10)
where $K_{ff}$ is the global stiffness matrix of body $j$ associated with the elastic coordinates. Solving Eq. (10) for $\frac{\partial q_f^j}{\partial X_e}$ gives

$$\frac{\partial q_f^j}{\partial X_e} = -K_{ff}^{-1} \frac{\partial K_{ff}}{\partial X_e} q_f^j(X_e)$$

(11)

Substituting Eq. (11) in Eq. (7) gives an expression for the sensitivity of the objective function, compliance, with respect to the design variable.

$$\frac{\partial C_j^l(X_e)}{\partial X_e} = -q_f^j(X_e) \frac{\partial K_{ff}}{\partial X_e} q_f^j(X_e),$$

(12)

where $\frac{\partial K_{ff}}{\partial X_e}$ is a known constant positive definite matrix. More detailed explanation is provided in [4].

### 2.3 Solid isotropic material with penalization approach in TOMBS

In topology optimization, the desired value of the design variable after convergence is either $X_e^{max} = 1$ or $X_e^{min} = \epsilon$, where, $\epsilon$ is a given threshold. Intermediate values need to be avoided. However, the intermediate values are always present. Solid isotropic material with penalization (SIMP) approach suggests penalization of the intermediate values such that more numbers of design variables reach the box limits after convergence. Penalizing is done by introducing an effective Young’s modulus $(X_e)^qE$, [2, 4]. This approach works for static response topology optimization where there is no mass in the system; however this kind of penalization is the reason of instability, mesh distortion and non-convergence of the optimization algorithm in the dynamic case regardless of the sensitivity analysis approach.

Element stiffnesses are proportional to Young’s modulus. Thus scaling the Young’s modulus is considerable when the design variable reaches small values in the optimization iteration steps. In flexible multibody model a uniform mass distribution is converted to a lumped mass distribution. The lumped masses are located in nodal points of the finite element mesh. Schematically the lumped masses are connected with springs shown in Figure 2. By reducing the stiffness of the elements around a lumped mass it will be no longer strongly attached to the body. Thus when the body experiences acceleration the mass does not follow the body’s trajectory. This is what is happening in TOMBS when the element stiffness is penalized in SIMP. In this case, the stiffness of an element might be different from the neighboring elements where the mass is the same. Hence, due to inertia force, elements with small design variable experience higher displacement than others. Here, such an element with high displacement is called a flying element, Figure 3. Consequently, the objective function shows a peak at the position of flying elements and in the next iteration step larger thickness is assigned to them giving raise to convergence failure of the process.

![Figure 2. A uniform mass distribution is converted to a lumped mass distribution.](image)
A simple modification of traditional SIMP reduces the effect of flying elements considerably. Here, in addition to the scaling Young’s modules it is suggested to scale element mass by scaling element’s density.

\[ E_j^{\ast} = (X_j^{\ast})^q E_j \quad \text{and} \quad \rho_j^{\ast} = (X_j^{\ast})^q \rho_j \]  

(13)

where, \( E_j \) is the Young’s modulus of the body material, \( E_j^{\ast} \) is the penalized Young’s modulus of an element, \( \rho_j \) is the density of the body, \( \rho_j^{\ast} \) is the penalized element density and \( q \) is the penalization factor. However, according to the nonlinear differential algebraic equation of motion of a flexible MBS Eq. (2), the relation between the lumped (or element) masses and element stiffness is not linear, thus, scaling shown in (13) might be not the best suggestion; on the other hand, this modification helps convergence of the optimization algorithm with no need of element removal.

3 Examples

In a flexible multibody system rigid bodies can also be present. However the optimization is done only on flexible bodies. One or several flexible bodies can be optimized simultaneously. Thus, the overall system behavior is accounted for during optimization process. A design space is assigned to the flexible bodies which are to be optimized. Then, material is removed from the body iteratively by the optimizer. We present here two examples to illustrates the pros and cons of the method. In both examples, the penalization factor is chosen to be \( q = 3 \) and the volume change is constrained by 40% of the initial volume. Also, four-node rectangular linear elements are used for finite element mesh of all flexible bodies here.

3.1 Slider–Crank

The first example is a simple slider–crank mechanism. First, the crank is driven with a constant angular velocity from zero angle, initial position of the crank, to 120° shown in Figure 4. TO is applied on both bodies simultaneously, cf. Figure 5 (left). The objective function and volume constraint history of both bodies as well as the elastic deformation of the upper center node of the connecting rod for a non-optimized design and the optimized one is illustrated in Figure 6. The non-optimized design is similar to Figure 4 but the thickness is changed such that the overall weight equals to the optimized design. The crank’s angular velocity is 2 rad/s and the mass of the slider is 2 kg. The boundary conditions are clamped-free and simply-supported for crank and connecting rod, respectively. The modulus of elasticity, density, thickness for both bodies are 70 GPa, 2700 kg/m³ and 0.02 m. A mass proportional damping proportional with constant 10 is introduced to the system. The convergence criterion is \( ||X^{(k+1)} - X^k|| < 0.085 \). Other TOMBS input data is provided in Table 1.

The average of the time dependent displacement field and similarly the average of the compliance varies with the time integration interval. Thus, the optimal design also depends on the time integration. To illustrate such a dependence the same slider–crank mechanism is optimized when the crank makes a complete revolution, Figure 5 (right). However, in this example, since the behavior...
of the system can be completely determined by only simulating one loop, it is possible to eliminate the dependency of the final topology on the integration interval. We compare two choices of integration interval, one (left figure) which contains all deflection cases of relevance only once, while the other covers an entire revolution and weights those deflections which last longer unrealistically higher (right).

If the convergence criterion is satisfied for one body, but not for the other one, its thickness is not updated at the next iteration step while the optimization process continues for the other body.

The reason for having a peak in the objective function history is that we do a constraint optimization. The optimizer tries to satisfy the volume constraint in a couple of first iterations where there is a jump in objective function history.

<table>
<thead>
<tr>
<th></th>
<th>Size[m]</th>
<th>Mesh size</th>
<th>Number of modes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conrod</td>
<td>$0.3 \times 0.03$</td>
<td>$150 \times 30$</td>
<td>3</td>
</tr>
<tr>
<td>Crank</td>
<td>$0.1 \times 0.03$</td>
<td>$100 \times 24$</td>
<td>3</td>
</tr>
</tbody>
</table>

### 3.2 Seven Body mechanism

The second example is a seven body mechanism, [15], with a constant driving angular velocity. A schematic is shown in Figure 7, where a design space is assigned to the each body which is exposed to topology optimization. First, we let Body 3 be the only flexible body in the system. The result of topology optimization is shown in Figure 8 (left).
Figure 6. (upper left) the objective function history of the connecting rod; (upper right) the objective function history of the crank; (lower left) the history of the volume constraint; (lower right) the elastic deformation of the upper center node of the connecting rod in one crank rotation.

Figure 7. (left) a schematic of a seven-body mechanism; (right) a design spaced is assigned to the bodies subject to topology optimization.

The time history of the displacement field is here the only input to the optimizer. If more bodies in the system are considered to be flexible the time history of the displacement field of Body 3 changes and thus the optimized design changes accordingly. This argument demonstrates the significance of the overall system behavior on the optimization process. Figure 8 (right) shows the optimal design of the system where topology optimization is applied on three bodies simultaneously. The objective function history for all three bodies is shown in Figure 9. The driver rotates with the speed of 1000 rad/s. The integration covers one complete loop of the driver. Simply-supported boundary conditions are used for all flexible bodies. The modulus of elasticity, density, thickness of flexible bodies are 70 GPa, 2700 kg/m³ and 5 × 10⁻³ m. The convergence criterion is
\[ \| f^{(k+1)} - f^k \| < 5 \cdot 10^{-6} \]. Other TOMBS data can be found in Table 2.

**Figure 8.** (left) the result of TOMBS on Body 3, where other bodies in the system are rigid; (right) optimal design where three bodies are optimized simultaneously.

**Figure 9.** left to right, the objective function history of bodies 3, 5 and 7.

<table>
<thead>
<tr>
<th></th>
<th>Size [mm]</th>
<th>Mesh size</th>
<th>Number of modes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Body 3</td>
<td>20 × 36</td>
<td>70 × 120</td>
<td>10</td>
</tr>
<tr>
<td>Body 5</td>
<td>40 × 9</td>
<td>150 × 40</td>
<td>3</td>
</tr>
<tr>
<td>Body 7</td>
<td>40 × 5</td>
<td>150 × 30</td>
<td>4</td>
</tr>
</tbody>
</table>

**Table 2.** TOMBS data for seven-body system.

### 3.3 Remarks
- The choice of the boundary condition of bodies influences the optimization result significantly.
- Static response topology optimization as well as TOMBS is sensitive with respect to optimization parameters such as filters, number of design variables and SIMP parameters. In addition, TOMBS is sensitive to MBS simulation parameters which might alter the displacement field such as number of considered modes, simulation interval and also some parameters that influence the differential algebraic equations solver performance such as absolute and relative tolerances.
- The time integration intervals must be chosen such to include all major deflections of bodies during the operation. If the MBS behaves periodically, at least one period can be enough. Weighted integration is an alternative.

### 4 CONCLUSIONS
We presented an implementation of topology optimization based on the dynamic behavior of an entire multibody system. We discussed simplifying assumptions on the sensitivity matrices, which
enabled us to achieve convergence of the optimization algorithm within reasonable computational time. These assumptions proved to be applicable in those cases where no peak accelerations occur. Furthermore we demonstrated the influence of the simulation time horizon on the optimization results.

REFERENCES


Structure preserving optimal control of a 3d-dimensional upright gait

Michael W. Koch*, Sigrid Leyendecker#

* Institute of Applied Mechanics
Technische Universität Kaiserslautern
D-67653 Kaiserslautern, Germany
koch_m@rhrk.uni-kl.de

# Chair of Applied Dynamics
University of Erlangen-Nuremberg
D-91052 Erlangen, Germany
Sigrid.Leyendecker@ltd.uni-erlangen.de

ABSTRACT
The optimal control of human locomotion requires simulation techniques, which handle the contact's establishing and releasing between the foot and the ground. In this work, our aim is to optimally control the human upright gait using a structure preserving variational integrator, whereby the physiologically motivated cost functions of minimal kinetic energy is chosen and the obtained results are analysed with the gait of humans. Thereby, the three-dimensional rigid multibody system enables us to model forefoot as well as heel contact and its dynamics is simulated using a structure preserving method. The applied mechanical integrator is based on a discrete constrained version of the Lagrange-d'Alembert principle, which yields a symplectic momentum preserving method (see [13] for details). The investigated contact formulation covers the theory of perfectly plastic contacts. To guarantee the structure preservation and the geometrical correctness, the non-smooth problem is solved including the contact configuration, time and force, in contrast to relying on a smooth approximation of the contact problem via a penalty potential.

Keywords: structure preservation, perfectly plastic contact, bipedal upright gait, optimal control.

1 INTRODUCTION
The human environment consists of a large variety of mechanical and biomechanical systems, in which different types of contact can occur. The biomechanical literature is often focussed on the function and structure of the human locomotor system in combination with the foot-ground contact, whereby cyclic walking movements come to the fore [7, 17]. Here, we are interested in the upright gait in conjunction with a structure preserving integrator. In contrast to movements with rolling wheels, the simulation of locomotion with legs requires the knowledge, how the contact’s establishing and releasing between the feet and the ground works. The investigated contact formulation covers the theory of perfectly plastic contacts (e.g. see [10]), which means that the foot stays in contact with the ground for a certain time.

The reduction to an armless bipedal walker represents the minimal multibody system to simulate humanlike upright gait. The walker consists of an upper body representing the human torso and each leg consists of three rigid bodies, which represent thigh, calf and the foot. The inclusion of the feet leads to movements that differ from those considered e.g. in [7]. The optimally controlled walker allows actuation in the hip and the knee joints as well as in the two ankle joints, such that a physiologically motivated cost function is minimised. In the numerical solution, a direct transcription method is used to transform the optimal control problem into an optimisation problem being constrained by the fulfilment of discrete equations of motion, boundary conditions and path constraints, see e.g. [12, 21]. The walking motions are subdivided in two phases and they are called single and double support phases. To avoid an artificial restriction of the optimisation problem’s phases by prescribing the time of contact establishing or releasing, variable time steps are used, wherefore the necessary scaling parameters are of the optimisation parameters.

The straight posture of the homo sapiens’ gait is a characteristical attribute of the human species, whereby the evolution of the human gait results from an anthropological optimisation process. Initiated by climbing down from trees and leaving forests, the survival in velds necessitates a physical adoption and the results are reflected by the kind of human locomotion respectively by
the physique: an upright gait shows benefits like a distinct all-round visibility, it reduces the water loss as a consequence of evaporation and allows the possibility to use tools and weapons at the struggle of survival. In paleoanthropology, the upright gait is appreciated as a key event of the human evolution with great changes of the anatomy: the human skeleton is optimised for bipedal walking with the result of an efficient and economical locomotion.

In biomechanical literature, the focus is often on analysing and simulating the upright gait with muscle models, whereas in robotics the aim is to develop bipedal robotics with the humanlike capability to move in various circumstances. In the area of computer graphics, a specific challenge is to create realistic movements for the virtual characters in video games or movies. According to the different research interests, a large variety of models exists to analyse bipedal walking. The models in the field of biomechanics range from a simple spring mass system [9] – explaining the basic dynamics of walking and running – to complex multibody systems with included muscles in [8]. These models are primarily used to investigate the acting forces within the body, but not to generate motions. In computer graphics, a lot of research is addressed to synthesise plausible motions and a further interesting point is, that in [19, 18] optimisation techniques find a transition from a pre-recorded motion to another one.

The aim of this paper is to compute physically valid and humanlike walking movements by using the DMOCC approach, which means that the dynamics of the multibody system is discretised by the variational integrator introduced in [13, 14] and the resulting equations of motion serve as equality constraints for the optimisation problem in which physiologically motivated cost function of minimal kinetic energy is tested. In contrast to the forward respectively inverse dynamic simulations, neither the trajectory nor the control sequence has to be exactly known for the walking system, because an optimal trajectory and the corresponding actuation is determined by the optimisation process subject to the evaluated objective function. Walking differs from running by the feet’s contact sequence: walking is characterised by switching from single and double support phases while running movements alternate between single support and flight phase. Often, in simpler models the double support phase is ignored, but herein it is an essential part of the following motion sequence. Another aspect concerns the three different contact possibilities of each foot during the walking motion: namely, we have to differentiate between forefoot, heel and double contact. Finally, the optimisation of a humanlike bipedal walking requires the correct consideration of the single and double support phase and also the different contact scenarios for each foot.

Section 2 describes briefly the multibody formulation in redundant coordinates and introduces a corresponding actuation force formulation. The symplectic momentum integrator and the null space method with nodal reparametrisation, which reduces the numerical effort, are introduced in Section 3. Section 4 covers the optimal control problem and explains shortly the transfer into a finite dimensional optimisation problem. The used bipedal walker model is described in Section 5 and the discrete equations of motion corresponding to the perfectly plastic contact for the variational approach are given. In Section 6, the general human gait sequence is described and in Section 7, the discrete constrained optimisation problem is formulated. The result for the bipedal walking with minimal kinetic energy is presented at the end of the paper.

2 RIGID MULTIBODY CONFIGURATION AND ACTUATION

In this work, the rotation free formulation introduced in [3] for rigid bodies and in [5] for rigid multibody systems is used to describe the configuration and to simulate the dynamics. The α-th rigid body is specified by a configuration vector \( \mathbf{q}^\alpha(t) \in \mathbb{R}^{12} \) composed by the placement of its center of mass \( \mathbf{q}^\alpha(t) \) and the right-handed director triad \( \mathbf{d}^\alpha_i(t) \) for \( i = 1, 2, 3 \). The director triad specifies the body’s orientation in space and has to stay orthonormal during the motion in the considered time interval \( [t_0, t_N] \), which is guaranteed by six so-called internal constraints \( \mathbf{g}_{\text{int}}(\mathbf{q}^\alpha) = 0 \in \mathbb{R}^6 \). In multibody systems, the rigid bodies are interconnected by different types of joints, e.g. revolute or spherical joints. The interconnection of the rigid bodies as well as their rigidity gives rise to a
scleronomic and holonomic constraint function \( g(q) \in \mathbb{R}^m \) on the redundant configuration variable \( q \in \mathbb{R}^k \), where \( k \) equals 12 times the number of bodies. The multibody systems are actuated directly by the independent generalised forces and torques \( \tau \in \mathbb{R}^{k-m} \) and the resulting \( k \)-dimensional redundant actuation \( f(q) \in \mathbb{R}^k \) can be computed via \( f(q) = B^t(q) : \tau \) with the input transformation matrix \( B^t(q) \in \mathbb{R}^{k \times (k-m)} \). Note that the transformation matrix depends on the rigid bodies’ interconnection and it is described in detail in [13].

### 3 STRUCTURE PRESERVING INTEGRATION FOR CONSTRAINED MECHANICAL SYSTEMS

The dynamics of time-continuous mechanical systems can be described using the Lagrangian or Hamiltonian formalism – in this work, the discrete Lagrangian mechanics is used to derive a structure preserving integrator, see e.g. [16]. The constrained mechanical system is considered in a configuration manifold \( Q \subseteq \mathbb{R}^k \) with the time-dependent configuration vector \( q(t) \in Q \). Corresponding to the approach in [13], the constrained version of the Lagrange-d’Alembert principle is discretised at the time nodes \{\( t_0, t_1 = t_0 + \Delta t, \ldots, t_n = t_0 + n \Delta t, \ldots, t_N = t_0 + N \Delta t \}\}, where \( N \in \mathbb{N} \) is the number of time intervals and the discrete configurations \( q_n \approx q(t_n) \) approximate the continuous trajectory. Similarly, \( \lambda_n \approx \lambda(t_n) \) approximates the Lagrange multipliers \( \lambda(t) \in \mathbb{R}^m \). As usual in the context of discrete variational mechanics, the discrete Lagrange-d’Alembert principle requires stationarity of the resulting action sum, i.e.

\[
\delta S_d = \delta \sum_{n=0}^{N-1} \mathcal{L}_d(q_n, q_{n+1}) - \frac{1}{2} (t_{n+1} - t_n) \left[ g^T(q_n) : \delta \lambda_n - g^T(q_{n+1}) : \delta \lambda_{n+1} \right] \\
+ \sum_{n=0}^{N-1} f^- \cdot \delta q_n + f^+ \cdot \delta q_{n+1} = 0
\]

for all variations \( \delta q_n \) with \( \delta q_0 = \delta q_N = 0 \) and \( \delta \lambda_n \). This leads to the \((k+m)\)-dimensional constrained forced discrete Euler-Lagrange equations

\[
D_2 \mathcal{L}_d(q_{n-1}, q_n) + D_1 \mathcal{L}_d(q_n, q_{n+1}) - G^T_d(q_n) : \lambda_n + f^+_{n-1} + f^-_n = 0 \\
g(q_{n+1}) = 0
\]  \( (1) \) \( (2) \)

for \( n = 1, \ldots, N-1 \). Here \( G_d = \frac{1}{2} (t_{n+1} - t_n - 1) \frac{\partial g(q_n)}{\partial q} \) denotes the \((m \times k)\)-dimensional Jacobian matrix of the constraints, and \( f^-_n = \frac{1}{2} (t_{n+1} - t_n) B^t(q_n) : \tau_n \), respectively \( f^+_{n-1} = \frac{1}{2} (t_n - t_{n-1}) B^t(q_n) : \tau_{n-1} \) are called left and right discrete forces. The resulting mechanical integrator represents exactly the behaviour of the analytical system concerning the consistency of the momentum maps and symplecticity. Due to these preservation properties it is called symplectic momentum scheme. A further benefit of this mechanical integrator is the good energy behaviour, which means that there is no numerical gaining or dissipation of energy.

According to [4, 5], we apply the discrete null space method to reduce the dimension of the constrained forced discrete Euler-Lagrange equations. The discrete null space matrix \( P \in \mathbb{R}^{k \times (k-m)} \) fulfils the property \( G_d \cdot P = 0 \) and premultiplying Equation (1) by the transposed null space matrix, the constraint forces and thereby the Lagrange multipliers vanish. The resulting \( k \)-dimensional system is called reduced forced discrete Euler-Lagrange equations. The minimal dimension of the system can be achieved using the vector of incremental generalised coordinates \( u_{n+1} \in U \subseteq \mathbb{R}^{(k-m)} \) to reparametrise the configuration vector \( q_{n+1} \) in the neighbourhood of \( q_n \). The nodal reparametrisation function \( F_d : U \times Q \rightarrow Q \)

\[
q_{n+1} = F_d(u_{n+1}, q_n)
\]  \( (3) \)

fulfils the constraint conditions and therefore Equation (2) becomes unnecessary. Finally, the number of unknowns and thereby the numerical effort is reduced by the formulation in discrete
generalised coordinates \( u_d = \{ u_d \}_{n=0}^N \) and the discrete torques \( \tau_d = \{ \tau_n \}_{n=0}^{N-1} \). The dimension of the equations of motion is reduced to \( k - m \).

\[
P^T(q_n) \cdot \left[ D_2 \mathcal{L}_d(q_{n-1}, q_n) + D_1 L \mathcal{L}_d(q_n, \mathbf{F}_d(u_{n+1}, q_n)) + f^+_n(q_n, \tau_{n-1}) + f^-_n(q_n, \tau_n) \right] = 0 \quad (4)
\]

4 OPTIMAL CONTROL PROBLEM

In general, the goal of optimal control problems is to determine the optimal state trajectory and force field for a holonomically constrained system, which moves from the initial state \( \mathbf{q}(t_0) = \mathbf{q}_0 \), \( \dot{\mathbf{q}}(t_0) = \dot{\mathbf{q}}_0 \) to a final state \( \mathbf{q}(t_N) = \mathbf{q}_N \). The investigated system fulfils the equations of motion and at the same time the objective functional

\[
J(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{f}) = \int_{t_0}^{t_N} C(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{f}) \, dt
\]

is minimised, where the integrand \( C(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{f}) : T Q \times T_q^* Q \to \mathbb{R} \) is a given cost function. The optimal control problem is solved using a direct transcription method, which transforms it into a constrained optimisation problem. The discrete objective function approximates the integral of the continuous cost function and the discrete constrained optimisation problems reads

\[
\min_{u_i, \tau_d} J(u_d, \tau_d) = \min_{u_i, \tau_d} \sum_{n=0}^{N-1} C(u_n, u_{n+1}, \tau_n),
\]

subject to the constraints given by the reduced discrete equations of motion of the symplectic momentum scheme in Equation (4). In addition to the discrete equations of motion of the specific mechanical integrator, further constraints, like initial conditions, final conditions and possible equality and inequality path constraints can be imposed.

5 HUMANOID BIPEDAL WALKER

The simulation of humanoid walking requires an adequate reproduction of the human locomotor system, whereby especially the movement abilities at the hip, knee and ankle joint are absolutely necessary to generate a natural gait.

It is a well known that pelvis, vertebrae as well as swinging arms are elements of human walking [2, 6], but their dynamic is summarised here at a torso representing rigid body. At this point it should be mentioned, that the swinging arms of the opposite side of the body in respect to the lower limbs reduce the angular momentum of the human body and enables a better balancing of the rotational motion as a result of the walking motion. The aim of this model reduction is to strictly
investigate the dynamics of the lower extremities with respect to the cost function in conjunction with a realistic contact establishing and releasing sequence. Therefore we decided to reduce our model to a minimal level of detail, which enables us to effectively investigate and evaluate the described research goals.

The human upright gait is analysed by using an elementary model consisting of seven rigid bodies, whereby each of the legs is modelled by three rigid bodies as illustrated in Figure 1. The right and left leg are absolutely identical to each other and the hips and ankles are modelled as spherical joints. Thigh and calf are connected via revolute joints, where the unit vectors $\mathbf{n}^2$ and $\mathbf{n}^5$ in the right and left thigh represents the axis of rotation. Consequently, the locomotor system consists of six connected rigid bodies and a seventh body summarises the human torso. The constrained multibody system is described by the configuration variable $\mathbf{q} \in \mathbb{R}^{84}$, composed by the placement of the rigid bodies’ center of mass and their right-handed director triad. As a result of the rigid body formulation, $m_{\text{int}} = 42$ internal constraints are present. The consideration of the anatomical joints causes $m_{\text{ext}} = 22$ and therefor the $k = 84$-dimensional system is restricted by 64 holonomic constraints. Corresponding to the $k - m = 20$ degrees of freedom, the generalised coordinates of the seven link model read

$$\mathbf{u} = \begin{bmatrix} \mathbf{u}^1 & \mathbf{\theta}^1 & \mathbf{\theta}^{(S)R} & \mathbf{\theta}^{(R)R} & \mathbf{\theta}^{(S)L} & \mathbf{\theta}^{(R)L} & \mathbf{\theta}^{(S)L} \end{bmatrix} \in \mathbb{R}^{20},$$

whereby $\mathbf{u}^1$ and $\mathbf{\theta}^1$ represent the translation respectively the rotational motion of the upper part of the human body. The walker model is only actuated in the hip, knee and ankle joints of both legs by

$$\mathbf{t}_{\text{LM}} = \begin{bmatrix} \mathbf{t}^R_H & \mathbf{t}^R_K & \mathbf{t}^L_H & \mathbf{t}^L_K & \mathbf{t}^L_A \end{bmatrix} \in \mathbb{R}^{14},$$

which means, that only internal actuation torques are applied. To allow the different contact scenarios during the walking sequence, two contact points are necessary to model the two single contact phases (forefoot contact (FC), respectively heel contact (HC)) as illustrated in Figure 2 and the complete contact support phase, at which the forefoot and the heel are in contact with the ground. Table 1 contains the physical quantities of the seven-link walker model and they are identical to the quantities for the jumper models [11], whereby $\mathbf{\rho}^R_{HC}$ and $\mathbf{\rho}^L_{FC}$ points from the centre of mass to the heel respectively forefoot contact point and vector $\mathbf{\rho}^R_A$ points to the ankle. In contrast to the jumper models, at which the thigh is fixed at the bottom of the torso, the right leg is connected with the upper part of the body at $\mathbf{\rho}^R_H = \begin{bmatrix} 0.0000 & -0.0936 & -0.2398 \end{bmatrix}^T \text{m}$ and due to the bilateral symmetry, the left thigh is fixed at $\mathbf{\rho}^L_H = \begin{bmatrix} 0.0000 & 0.0936 & -0.2398 \end{bmatrix}^T \text{m}$. 

<table>
<thead>
<tr>
<th>physical quantity</th>
<th>torso</th>
<th>thigh</th>
<th>calf</th>
<th>foot</th>
</tr>
</thead>
<tbody>
<tr>
<td>mass [kg]</td>
<td>33.9946</td>
<td>6.5233</td>
<td>2.6857</td>
<td>0.8372</td>
</tr>
<tr>
<td>moment of inertia $I_{ee_1}$ [kg·m²]</td>
<td>1.6194</td>
<td>0.1137</td>
<td>0.0391</td>
<td>0.0034</td>
</tr>
<tr>
<td>$I_{ee_2}$ [kg·m²]</td>
<td>1.0876</td>
<td>0.1158</td>
<td>0.0393</td>
<td>0.0030</td>
</tr>
<tr>
<td>$I_{ee_3}$ [kg·m²]</td>
<td>0.3785</td>
<td>0.0225</td>
<td>0.0029</td>
<td>0.0007</td>
</tr>
<tr>
<td>length [m]</td>
<td>0.6644</td>
<td>0.4582</td>
<td>0.3753</td>
<td>0.2433</td>
</tr>
<tr>
<td>center of mass from proximal centroid [m]</td>
<td>0.3470</td>
<td>0.1779</td>
<td>0.1943</td>
<td>0.1368</td>
</tr>
</tbody>
</table>

Table 1: Physical quantities of the human torso, thigh, Table 2: Geometrical quantities of the feet on the basis of the results from [1].
6 HUMAN GAIT

This section describes the sequence of human gait as multiphase problem based on a multibody system and the achieved forward locomotion as a result of the lower extremities’ movement (compare to [15]). The human gait is defined as bipedal, forward propulsion of the human body’s centre of gravity and during the motion, alternate sinuous movements of different body parts are observable. The different gait patterns are the result of differences in the movements of the lowers extremities concerning velocity, forces, kinetic and potential energy and especially the changes of contact between foot and ground. The so-called natural gaits are classified as meander, walk, jog, run and sprint with increasing order of speed [20]. Generally, the natural gaits are designed to propel a person in forward direction, but they can also be adopted for lateral movement.

We are interested in a straight walking movement and it is assumed that the right and left steps are identical mirror images of each other. Therefor, only half a gait cycle is optimised with periodicity constraints on the posture, the velocities and actuation can be defined. Analogous to the compass gait in [14], a mirror function in conjunction with a mirror plane is introduced to guarantee the final state is a mirrored image of the initial state and it is displaced in walking direction, compare the left and right photo of Figure 3. The mirror plane is placed in the middle between the heel contact points and it is spanned by the vertical unit vector and the walking direction. Let $\gamma_{LM}^{mirr} : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ denotes the mirror function of the bipedal walker, then it is possible to model the periodicity constraints for the legs as mirror images of the configurations and control variables in the form

\[
\begin{align*}
\mathbf{d}_0^j &= \gamma_{LM}^{mirr}(\mathbf{d}_{b_0}^{j+3}), \quad \mathbf{d}_1^j = \gamma_{LM}^{mirr}(\mathbf{d}_{b_1}^{j+3}) \quad \text{for} \quad j = 2, 3, 4; \quad (7) \\
\mathbf{d}_0^j &= \gamma_{LM}^{mirr}(\mathbf{d}_{b_0}^{j+3}), \quad \mathbf{d}_1^j = \gamma_{LM}^{mirr}(\mathbf{d}_{b_1}^{j-3}) \quad \text{for} \quad j = 5, 6, 7; \quad \text{and} \quad (8) \\
\tau_0^j &= \gamma_{LM}^{mirr}(\tau_{N-1}^j), \quad \tau_0^j = \gamma_{LM}^{mirr}(\tau_{N-1}^j). \quad (9)
\end{align*}
\]

At this point it should be mentioned, that the periodic boundary conditions can also be realised by mirror images of the leg’s directors and by the conjugate momenta at the boundary time nodes instead of two director conditions. According to the illustrated sequence of forehead and heel contact realising and establishing in Figure 3, the human gait motion has to be adapted in a correct

Figure 3: Photographies of human walk cycle for a single step and illustration of the support phases.
order in the continuous optimal control problem as well as in the discrete optimisation problem. In both cases, the double support phase represents the essential challenges of the human walking phases. Comparing with running, the walking movement is characterised by the chronology of single and double support phases and thus at every time at least one foot is in contact with the ground: no flight phase occurs as observable e.g. in running motions.

7 OPTIMAL CONTROL OF THE BIPEDAL JUMPER

The constrained optimisation problem is formulated in terms of generalised coordinates \( 7LM \mathbf{u}_d \) and the actuation torques \( 7LM \mathbf{\tau}_d \). In case of free motion (no contacts occur), the dynamical system is restricted by the internal and external constraints in order to model the rigidity of the bodies and the structural composition of the multibody system itself. The dynamic of the seven-link model is described by a 20-dimensional system of equations. In all five walking sections, the same relative nodal reparametrisation is used to update the configuration in dependence of the generalised coordinates. The equations of motion during the walking phases are premultiplied with the appropriate forefoot respectively full contact discrete null space matrices to eliminate the constraint forces including the contact forces. The equations of motion are premultiplied by the corresponding null space matrix to achieve a reduced dimension of the equation of motions. At the end of the optimised walking phase, the left leg is in full contact with the ground. The contacts between the feet and the ground are modelled as perfectly plastic contact and the orientation of the contact force prevents penetrating the ground. As a result of the different contact scenarios during the walking phases, the Lagrange multipliers are determined in post-processing via

\[
\lambda_{C_n} = R^T(q_n) \cdot \left[ D_2 L_d(q_{n-1}, q_n) + D_1 L_d(q_n, q_{n+1}) + f^+_n + f_n^- \right],
\]

where

\[
R^\Gamma(q_n) = 7LM G^T_\Gamma(q_n) \cdot (7LM G_\Gamma(q_n) \cdot 7LM G^T_\Gamma(q_n))^{-1} \quad \text{for} \quad \Gamma = \text{FHC, HC, FC},
\]

which is formulated with the discrete Jacobian \( R^\Gamma(q_n) \) of the seven-link model. During the double support phases, the Coulomb’s static friction law has to be fulfilled for the feet and consequently two inequality constraints are necessary.

The determination of the respective null space matrices for the double support phase is avoided by taking a similar approach as for the transfer of contact at the compass gait. The first double support phase is characterised by the forefoot contact of the right leg and by the heel contact of the left leg. At first, the Euler-Lagrange equations are premultiplied by the transposed discrete null space matrix \( R^T \mathbf{P}_{FC} \) to eliminate the contact forces at the right foot, afterwards we premultiply the partially reduced equations of motion again by the following projection matrix

\[
\mathbf{Q}_{HC} = I - \left( L^\top \mathbf{G}_{HC} \cdot R \mathbf{P}_{FC} \right) \cdot \left( \left( L^\top \mathbf{G}_{HC} \cdot R \mathbf{P}_{FC} \right) \cdot \left( L^\top \mathbf{G}_{HC} \cdot R \mathbf{P}_{FC} \right)^\top \right)^{-1} \cdot L^\top \mathbf{G}_{HC} \cdot R \mathbf{P}_{FC} \in \mathbb{R}^{17 \times 17}.
\]

At the second double support phase – forefoot contact on the right foot and full contact on the left leg – the equations of motion are premultiplied at first by \( L^T \mathbf{P}_{FC} \) and then by the matrix

\[
\mathbf{Q}_{HFC} = I - \left( L^\top \mathbf{G}_{HFC} \cdot L \mathbf{P}_{HC} \right) \cdot \left( \left( L^\top \mathbf{G}_{HFC} \cdot L \mathbf{P}_{HC} \right) \cdot \left( L^\top \mathbf{G}_{HFC} \cdot L \mathbf{P}_{HC} \right)^\top \right)^{-1} \cdot L^\top \mathbf{G}_{HFC} \cdot L \mathbf{P}_{HC} \in \mathbb{R}^{17 \times 17}.
\]

Due to the unknown switching times between the different walking sequences, four scaling parameters summarised in \( \sigma = [\sigma_1 \ \sigma_2 \ \sigma_3 \ \sigma_4]^T \in \mathbb{R}^4 \) are required and they are also part of the optimisation variables. It is necessary to mention, that the scaling parameter \( \sigma_3 \) is used twice, namely at the single support phases at the beginning and at the end of the optimisation problem. This parameters enables the optimiser to shorten or extend each walking phase inside the scaling limits. Finally, the discrete constrained optimisation problem of the half cycle gait for the chosen conditions reads

\[
\min_{\mathbf{u}_d, \mathbf{\tau}_d, \sigma} J(\mathbf{u}_d, \mathbf{\tau}_d, \sigma) = \sum_{n=1}^{N-1} C(\mathbf{u}_n, \mathbf{u}_{n+1}, \mathbf{\tau}_n, \mathbf{\tau}_{n+1}, L_n, L_{n+1})
\]
subject to

- reduced forced discrete equations of motion of the single support phase with full contact in 
  \([t_0, t_\xi]\) for the right foot \((\alpha = R)\) and in \([t_\kappa, t_N]\) for the left foot \((\alpha = L)\)
  \[ \alpha \mathbf{P}_F(q_n) \cdot [D_2 F_{d}(q_{n-1}, q_n) + D_1 F_d(q_n, q_{n+1})] = 0 \]
  \[ \alpha \mathbf{g}_{FHC}(q_{n+1}) = 0 \]

- constrained motion of the single support phase with forefoot contact in \([t_\xi, t_\kappa]\) for the right foot
  \[ R \mathbf{P}_F(q_n) \cdot [D_2 F_{d}(q_{n-1}, q_n) + D_1 F_d(q_n, q_{n+1})] = 0 \]
  \[ R \mathbf{g}_{FC}(q_{n+1}) = 0 \]

- constrained motion of the double support phase with forefoot contact of the right foot and full
  contact of the left foot in \([t_1, t_\xi]\)
  \[ L \mathbf{Q}_{HC} \cdot R \mathbf{P}_F(q_n) \cdot [D_2 F_{d}(q_{n-1}, q_n) + D_1 F_d(q_n, q_{n+1})] = 0 \]
  \[ R \mathbf{g}_{FC}(q_{n+1}) = 0 \]

- periodic boundary conditions of equation (7) – (9) are summarised by
  \[ \gamma^{LM} \mathbf{b}(q_0, q_1, q_{N-1}, q_N, \tau_1, \tau_{N-1}) = 0 \] (11)

- path constraints for \(n = 1, \ldots, N\)
  \[ \gamma^{LM} \mathbf{h}_{eq}(q_n) = 0 \]
  \[ \gamma^{LM} \mathbf{h}_{ineq}(q_n) < 0 \]

- path constraints single support
  - full contact for \(n = 1, \ldots, N_\xi - 1\) and \(n = N_\kappa + 1, \ldots, N - 1\)
    - right leg: \(n = 1, \ldots, N_\xi - 1\):
      \[ \sqrt{\left(\frac{\lambda^1_{FHC,n}}{\lambda^2_{FHC,n}}\right)^2 + \left(\frac{\lambda^3_{FHC,n}}{\lambda^4_{FHC,n}}\right)^2} - \mu_0 \lambda^5_{FHC,n} + \lambda^6_{FHC,n} < 0 \]
    - right leg: \(n = N_\xi\):
      \[ \lambda^4_{FHC,n} = \lambda^5_{FHC,n} = \lambda^6_{FHC,n} = 0 \]
    - left leg: \(n = N_\kappa + 1, \ldots, N - 1\):
      \[ \sqrt{\left(\frac{\lambda^1_{FHC,n}}{\lambda^2_{FHC,n}}\right)^2 + \left(\frac{\lambda^3_{FHC,n}}{\lambda^4_{FHC,n}}\right)^2} - \mu_0 \lambda^5_{FHC,n} + \lambda^6_{FHC,n} < 0 \]
  - forefoot contact for \(n = N_\xi + 1, \ldots, N_\iota - 1\)
    - right leg:
      \[ \sqrt{\left(\frac{\lambda^1_{FC,n}}{\lambda^2_{FC,n}}\right)^2 + \left(\frac{\lambda^3_{FC,n}}{\lambda^4_{FC,n}}\right)^2} - \mu_0 \lambda^5_{FC,n} < 0 \]
path constraints double support

- single/single contact for \( n = N_1, \ldots, N_\zeta - 1 \)

left foot:
\[
\sqrt{\left( \lambda_{HC,n}^1 \right)^2 + \left( \lambda_{HC,n}^2 \right)^2} - \mu_0 \| \lambda_{HC,n}^3 \| < 0
\]

right foot:
\[
\sqrt{\left( \lambda_{FC,n}^1 \right)^2 + \left( \lambda_{FC,n}^2 \right)^2} - \mu_0 \| \lambda_{FC,n}^3 \| < 0
\]

- single/full contact for \( n = N_\zeta, \ldots, N_\kappa \)

left leg: \( n = N_\zeta, \ldots, N_\kappa \):
\[
\sqrt{\left( \lambda_{FHC,n}^1 \right)^2 + \left( \lambda_{FHC,n}^2 \right)^2} - \mu_0 \| \lambda_{FHC,n}^3 + \lambda_{FHC,n}^6 \| < 0
\]

right leg: for \( n = N_\zeta, \ldots, N_\kappa - 1 \):
\[
\sqrt{\left( \lambda_{FC,n}^1 \right)^2 + \left( \lambda_{FC,n}^2 \right)^2} - \mu_0 \| \lambda_{FC,n}^3 \| < 0
\]

right \( n = N_\kappa \):
\[
\lambda_{FC,n}^1 = \lambda_{FC,n}^2 = \lambda_{FC,n}^3 = 0
\]

- scaling parameters
\[
\sigma_{LB} \leq \sigma \leq \sigma_{UB}.
\]

Each of the walking phases is scaled by its appropriate scaling parameter; remind that \( \sigma_1 \) is used twice, namely at the single support phases at the beginning and at the end. Analogous to the optimisation of the jumping movement in [11], the scaling parameters yield a lower and upper bound of the manoeuvre time for the half gait, while the optimiser manoeuvre time is determined by the optimiser.

### 7.1 Minimal kinetic energy per step length

The optimised motion for the criterion of Minimal kinetic energy per step length is presented in the following lines. In Figure 4, the trajectories of the human and the feet as well as some characteristic snapshots are given. Obviously, as the trajectory of the swinging leg indicates, the foot is only lifted up as little as possible – the feet’s center of mass is always below a height of 0.2 m (compare to Figure 4). The kinetic energy sum for a step-length of about 0.6338 m is computed to 41.2256 J s. The optimiser extends all walking phases according to \( \sigma = [1.5000 \ 1.5000 \ 1.4889 \ 1.4724] \), which is in all four cases exactly or almost the maximum scaling parameters and the walking motion takes 0.5988 s. Another interpretation of this cost-function is, that we are interested at the minimal velocity per step-length, because the kinetic energy is directly proportional to the walking speed. The feet’s trajectory over time is illustrated in Figure 5 and obviously, the comparatively large rotational motions at the angle joint can be observed. This can be explained by the fact that due to the small mass of the feet the resulting energy is small. Obviously, the standing leg is primarily actuated in \( e_2 \)-direction at the angle joint during the forefoot contact, compare to Figure 6.
The applied actuation torques at the hip and knees have a subordinate role. The calculated contact Lagrange multipliers are depicted in Figure 7 whereby also Coulomb’s static friction law is satisfied during the whole motion. Figure 8 depicts the evolution of the angular momentum and the energy during the optimised half gait cycle. Obviously, the different walking phases are recognisable at the evolution of the angular momentum, especially in component $L_2$. The right-hand side of Figure 8 shows the kinetic, potential as well as the total energy during the optimised gait cycle.

8 CONCLUSIONS

In this work, a variational integrator with structure preserving properties is used to investigate the bipedal upright gait in a optimal control simulation. The contact between the feet and the ground during the support phases is modelled by a perfectly plastic contact formulation. The optimal control problem of the human gait is solved by a direct transcription method to transform it into an optimisation problem, whereby the goal is to minimise the minimal kinetic energy per step length. Four scaling parameter are introduced to optimise the duration of the different phases. Finally, the solution of the optimisation problem reflects a realistic walking movement, which should in future also investigated for other physiologically motivated cost functions.

REFERENCES


Figure 6: Minimal kinetic energy per step length: actuation torques of the left and right leg during the half gait cycle.


Figure 7: Minimal kinetic energy per step length: evolution of the contact forces of the left and right leg during the half gait cycle.

Figure 8: Minimal kinetic energy per step length: evolution of the angular momentum (left) and the energy (right) during the half gait cycle.


Trajectory tracking problem for omnidirectional mobile robot with parameter variations and delayed feedback

Aleksandr S. Andreev*, Olga A. Peregudova#

*Faculty of Mathematics and Information Security
Ulyanovsk State University
Leo Tolstoi str., 42, 432017
Ulyanovsk, Russia
AndreevAS@ulsu.ru

#Faculty of Mathematics and Information Security
Ulyanovsk State University
Leo Tolstoi str., 42, 432017
Ulyanovsk, Russia
peregudovaoa@sv.ulsu.ru

ABSTRACT

This paper presents a continuous control law to solve a trajectory tracking problem for a mobile omnidirectional wheeled robot with three independent driving wheels. Robustness to parameter uncertainties and delay in the feedback is achieved using Lyapunov functions method. Experimental results are presented to show the usefulness of the proposed control.

Keywords: Wheeled mobile robot, omnidirectional, control, delay, Lyapunov function.

1 INTRODUCTION

An omnidirectional mobile robot is a complex, controllable electromechanical system. The problem of tracking control of such type of wheeled mobile robots has attracted a lot of attention over the past twenty years. The design of omnidirectional mobile robots assumes that the wheels have fixed rollers, the axes of rotation of which lie in the plane of the wheels. Thanks to these wheels the robot may move in any direction without turning, which greatly improves the maneuverability of the robot and can be used in confined spaces.

In [1, 2] the dynamic models describing the motion of the robot controlled by the action of DC motors are constructed and the problem of stabilization of stationary motions of the robot is considered. In [3] the method of linear motion control of such systems on the basis of PI and PD controllers is proposed. The studies on the development of new methods and algorithms for nonlinear control of mobile robots with omni-wheels that draw the robot on the desired trajectory and provide the motion stabilization along it, under conditions of incomplete information about the inertia characteristics of the robot and the action of external perturbations are of great interest. At the number of studies [4, 5] there is proposed the procedure of the backstepping method for constructing of adaptive and robust nonlinear motion controls in view of uncertain disturbances and unknown system parameters. In this case, an important issue is to take into account the delay in the control structure which may lead to loss of stability of the system. In the study of systems with delay some difficulties associated with the choice of Lyapunov function arise [6, 7]. The construction of the delayed control for mobile robots wheel is poorly known issue. In [8] a discrete control with constant delay for the kinematic model of the robot is proposed. This leaves open the question of the construction of control with variable delay for dynamic models.

In this paper we consider the problem of the trajectory tracking control of the mobile robot with three omnidirectional wheels with considering the varying control delay and the obscurity of mass of the platform. The problem is solved by using a continuous control law with saturation function. The control law is confirmed by numerical simulation.

2 PROBLEM FORMULATION
The equations of dynamic model of the robot with three independent driving omnidirectional wheels equally spaced at 120 degrees from one another (fig. 1) are as follows.

\[
\begin{align*}
(m + \Delta m)\dddot{\xi} + h\ddot{\xi} + m_j\dot{\psi}\dot{\eta} &= \sin \psi u_1 + \sin(\psi + \frac{2\pi}{3})u_2 + \sin(\psi + \frac{4\pi}{3})u_3 \\
(m + \Delta m)\dddot{\eta} + h\ddot{\eta} - m_j\dot{\psi}\dot{\xi} &= -\cos \psi u_1 - \cos(\psi + \frac{2\pi}{3})u_2 - \cos(\psi + \frac{4\pi}{3})u_3 \\
(I + \Delta I)\dddot{\psi} + 2a^2 h\dot{\psi} &= -a(u_1 + u_2 + u_3)
\end{align*}
\]

Here $\xi$ and $\eta$ are the coordinates of the center of a platform of the robot in Cartesian system $O\xi\eta\zeta$; $\psi$ is the platform angle of rotation round a vertical counted from an axis $\xi$; $u_1$, $u_2$, and $u_3$ are the control inputs given on electric motors; $a$ is a distance from the center of the platform to the center of each wheel; the constant $h$ is defined both by coefficient of the moment of the antielectromotive force and radius of a wheel; $m$, $I$ and $d$ are the known components of masso-inertial parameters of the system; $\Delta m$ and $\Delta I$ are the unknown components of mass of a platform and its moment of inertia respectively satisfying to restrictions: $|\Delta m| < \Delta m_0 = \text{const}$, $|\Delta I| < \Delta I_0 = \text{const}$, and $\Delta m_0 < m$, $\Delta I_0 < I$.

Figure 1. Model of the robot with three omnidirectional wheels.

We assume that at structure of feedback there is some unknown variable delay $\tau(t) < \tau_0 = \text{const} > 0$, which arises owing to delays in operation of devices of a control system and final speed of signaling on a network. Let $q_o(t) = (\xi_o(t), \eta_o(t), \psi_o(t))^T$ be some twice continuously differentiable limited function defined for all $t > -2\tau_0$.

The problem of trajectory tracking of the robot consists in the following. It is required to find a control $u(t - \tau(t))$ ( $u = (u_1, u_2, u_3)^T$, $|u| < u_0 = \text{const}$ ) and to specify the restrictions on parameters of the system and a trajectory, at which for some number $\epsilon > 0$ (tracking error) will be numbers $\delta > 0$ and $t^* > 0$, such that for any initial function $\varphi(s)$, $-2\tau_0 < s < 0$, satisfying to the inequality

\[
\max_{-2\tau_0 < s < 0} |(\varphi(s) - q_o(s), \dot{\varphi}(s) - \dot{q}_o(s))| < \delta
\]

for all solutions of system (1) with condition $q(s) = \varphi(s)$, $-2\tau_0 < s < 0$, for all $t > t^*$ the inequality $|q(t) - q_o(t)| < \epsilon$ will take place.
3 PROBLEM SOLUTION

To solve the trajectory tracking problem we will take control law \( u = u(t - \tau(t)) \) in the following form

\[
u = P^{-1}(y_j(t - \tau(t)) + \psi_o(t))(\text{diag}(h_{k_1}, h_{k_2}, 2a^2 h_{k_3})\text{sat}(t - \tau(t)) + g_o(t)) \tag{2}\]

Here the matrix \( P^{-1} \) is the inverse for the matrix of coefficients at the control \( u \) in the right side of (1); \( y_j = \psi - \psi_o(t) \); \( k_1, k_2 \) are coefficients which are selected according to the requirements of limited control and sufficient rate of convergence to the desired motion; vector function of saturation \( \text{sat}(z) = (\text{sat}(z_1), \text{sat}(z_2), \text{sat}(z_3))^T \) has the following components:

\[
\text{sat}(z_i) = \begin{cases} 
sign(z_i), & |z_i| \geq \gamma, \\
\frac{z_i}{\gamma}, & |z_i| < \gamma, \quad i = 1, 2, 3, \quad \gamma = \text{const} > 0
\end{cases}
\]

\[
z_1(t) = c \left( \dot{z}(t) - \dot{z}_0(t) + \frac{m(\dot{z}(t) - \dot{z}_0(t))}{h} \right), \quad z_2(t) = c \left( \ddot{z}(t) - \ddot{z}_0(t) + \frac{m(\ddot{z}(t) - \ddot{z}_0(t))}{h} \right),
\]

\[
z_3(t) = c \left( \psi(t) - \psi_o(t) + \frac{1(\psi(t) - \psi_o(t))}{2a^2 h} \right)
\tag{3}
\]

Here \( c > 1 \) is some number. The function of saturation with large angle of switching line (for a sufficiently small number \( \gamma \) ) is a continuous approximation for relay function. Vector function \( g_o(t) \) in the expression (2) has the following form

\[
g_o(t) = \left( m\dot{\psi}_0(t) + h\dot{\psi}_0(t) + m_d \dot{\psi}_0(t)\psi_o(t), m\ddot{\psi}_0(t) + h\ddot{\psi}_0(t) - m_d \dot{\psi}_0(t)\psi_o(t), 2a^2 h\psi_o(t) \right)^T
\]

We introduce the following notations

\[
q_1(t, x_1, x_2, x_3, x_4, x_5) = \frac{1 + c}{m - \Delta m_0} \left[ \frac{\Delta m_0 h}{m} \left( |\dot{\psi}_0(t)| + \frac{2a^2 m |\ddot{\psi}_0(t)|}{l} \right) \right] x_1 +
\]

\[
+ \frac{2(1 + c)^2 m_d a^2 h}{(m - \Delta m_0)lc} x_2 + \frac{ck, m \sin\left( \frac{2(1 + c)a^2 h\tau_o x_5}{lc} \right)}{m - \Delta m_0} x_3 - \frac{ck, m \cos\left( \frac{2(1 + c)a^2 h\tau_o x_5}{lc} \right)}{m + \Delta m_0} x_4 +
\]

\[
+ mc |\dot{\psi}_0(t)| \left( 1 - \frac{m}{m + \Delta m_0} \cos\left( \frac{2(1 + c)a^2 h\tau_o x_5}{lc} \right) \right) +
\]

\[
+ mc |h\dot{\psi}_0(t) + m_d \dot{\psi}_0(t)\psi_o(t)| \left( 1 - \cos\left( \frac{2(1 + c)a^2 h\tau_o x_5}{lc} \right) \right) +
\]

\[
+ \frac{m c |m\ddot{\psi}_0(t) + h\ddot{\psi}_0(t) - m_d \dot{\psi}_0(t)\psi_o(t)| \sin\left( \frac{2(1 + c)a^2 h\tau_o x_5}{lc} \right)}{h(m - \Delta m_0)} +
\]

\[
+ \frac{m c |m\dddot{\psi}_0(t) + h\dddot{\psi}_0(t) - m_d \ddot{\psi}_0(t)\dddot{\psi}_o(t)| \sin\left( \frac{2(1 + c)a^2 h\tau_o x_5}{lc} \right)}{h(m - \Delta m_0)}
\]
In our notations the following theorem on the tracking of a given trajectory of the robot holds.

**Theorem 1.** Suppose that there is the number \( c > 1 \) such that

\[
2(1 + c)a^2h\Delta I_0 < \frac{ck_z I(I - \tau_0 c k_z) - \Delta I_0 \gamma}{I(\gamma + \tau_0 c k_z + \Delta I_0 \gamma)}
\]

and the following conditions are hold

1) The numbers \( \delta_0 > \gamma_0 \) and \( \delta_1 > \gamma_1 \) exist such that

\[
\frac{2(1 + c)a^2h\Delta I_0 \delta_0}{Ic} < k_3, \quad q_i(t, \delta_1, \delta_0, \delta_1, 1, \delta_0) \leq -\varepsilon_1 = \text{const}, \quad i = 1, 2, \quad t \geq 0
\]

2) The following inequalities are hold

\[
2(1 + c)a^2h\Delta I_0 \gamma_0 < \frac{ck_z I(1 - \tau_0 c k_z) - \Delta I_0 \gamma}{I(1 + \tau_0 c k_z) + \Delta I_0}
\]
\[ q_i(t, \gamma_1, \gamma_0 \gamma_1, 1, 1, \gamma_0) + \frac{c k \tau_m \cos(2(1+c)a^2 h \tau_0 \gamma_0)}{l \gamma (m - \Delta m)} \max_{\gamma > 0} | q_i (s, \gamma_1, \gamma_0 \gamma_1, 1, 1, \gamma_0) | \leq - \epsilon_i \]

\[ \epsilon_i = \text{const} > 0, \quad i = 1, 2, \quad t \geq 0 \]

3) The positive number \( \epsilon < \gamma \) exists such that \( \forall t \geq 0 \) the following inequalities are true

\[ q_i \left(t, \epsilon, 0, \epsilon, \epsilon, 0\right) + \frac{c k \tau_m \max_{\gamma > 0} | q_i (s, \epsilon, \epsilon, \epsilon, \epsilon, \epsilon) | \leq - \epsilon_i \]

\[ \epsilon_i = \text{const} > 0, \quad i = 1, 2, \quad t \geq 0 \]

Then the control (2) solves the problem of tracking of the trajectory \( q_0(t) \) with tracking error which does not exceed of the number \( \epsilon \). The set of the initial perturbations \( \varphi(s) = (\varphi_1(s), \varphi_2(s), \varphi_3(s))^T \), \( -2 \tau_0 \leq s \leq 0 \), satisfies the following constraints

\[ \max \left\{ \max_{-2 \tau_0 \leq s \leq 0} | \varphi_1(s) - \tilde{\xi}_0(s) |, c \max_{-2 \tau_0 \leq s \leq 0} | \varphi_1(s) - \tilde{\xi}_0(s) | + \frac{m}{h} (\varphi_1(s) - \tilde{\xi}_0(s)) \right\} < \delta_i \]

\[ \max \left\{ \max_{-2 \tau_0 \leq s \leq 0} | \varphi_2(s) - \eta_0(s) |, c \max_{-2 \tau_0 \leq s \leq 0} | \varphi_2(s) - \eta_0(s) | + \frac{m}{h} (\varphi_2(s) - \eta_0(s)) \right\} < \delta_i \]

\[ \max \left\{ \max_{-2 \tau_0 \leq s \leq 0} | \varphi_3(s) - \psi_0(s) |, c \max_{-2 \tau_0 \leq s \leq 0} | \varphi_3(s) - \psi_0(s) | + \frac{I(\varphi_3(s) - \psi_0(s))}{2a^2 h} \right\} < \delta_0 \]

**Proof.** We introduce the following notations

\[ y = (y_1, y_2, y_3)^T, \quad z = (z_1, z_2, z_3)^T \]

where \( y_1 = \xi - \bar{\xi}_0(t) \), \( y_2 = \eta - \bar{\eta}_0(t) \), \( y_3 = \psi - \bar{\psi}_0(t) \) and the variables \( z_i \) \( (i = 1, 2, 3) \) are determined according to the relationships (3). Then in deviations \( (y, z) \) the system (1) with (2) can be written in the form

\[ \begin{cases} \dot{y} = -\text{diag} \left( \frac{h}{m}, \frac{h}{m}, \frac{2a^2 h}{1} \right) y + \frac{1}{c} \text{diag} \left( \frac{h}{m}, \frac{h}{m}, \frac{2a^2 h}{1} \right) z, \\ \dot{z} = -cA(t)y + A(t)z - cS(y, z) - \\ -P_1(y_3(t) - y_3(t - \tau(t)))c \text{diag} \left( \frac{k_m}{m + \Delta m}, \frac{k_m}{m + \Delta m}, \frac{k_m}{I + \Delta I} \right) \text{sat}(z(t - \tau(t)) - cS_1(t) \right) \end{cases} \]

(4)

Where

\[ A(t) = \begin{pmatrix} h \Delta m & -m \Delta m \psi_0(t) & -2a^2 m^2 \Delta m \eta_0(t) \\ -m \Delta m \psi_0(t) & h \Delta m & -2a^2 m^2 \psi_0(t) \\ 0 & 0 & 2a^2 h \Delta I \end{pmatrix} \]
We choose the Lyapunov vector-function \( V = V(y, z) \) for the system (4) in the form
\[
V = (V_1, V_2)^T, \quad V_1 = \max(|y_1|, |y_2|, |z_1|, |z_2|), \quad V_2 = \max(|y_3|, |z_3|)
\] (5)

For the more effective evaluation of the right derivative of the vector function (5) by virtue of (4) we transform the system (4) by applying the ratio
\[
\frac{ds}{dt} = \text{sat}(\sigma(t)) - r(t) \frac{d}{dt}\text{sat}(\sigma(t)) - \frac{dz}{dt}.
\] (6)

where the functions \( r(t) \) and \( \sigma(t) \) satisfy the inequalities: \( 0 \leq r(t) \leq \sigma(t) \leq \sigma(t) \leq t \).

We will take into account that \( \frac{ds}{dt} = 0 \) if \( |z_i| > \gamma \); \( \frac{ds}{dt} = \gamma^{-1} \) if \( |z_i| < \gamma \) (\( i = 1, 2, 3 \)). For \( |z_i| = \gamma \) we put \( \frac{ds}{dt} = 0 \). With conversion (6) the system (4) takes the following form
\[
\begin{align*}
\dot{y} &= -\text{diag}\left(\frac{h}{m}, \frac{h}{m}, \frac{2a^2h}{I}\right)y + \frac{1}{c} \text{diag}\left(\frac{h}{m}, \frac{h}{m}, \frac{2a^2h}{I}\right)z, \\
\dot{z} &= -cA(t)y + A(t)z - cS(t) - cG(y, z) - \\
- &P(t)(y_i(t) - y_i(t - \tau(t))) \text{diag}\left(\frac{k_m}{m + \Delta m}, \frac{k_m}{m + \Delta m}, \frac{k_I}{I + \Delta I}\right)\text{sat}(z(t)) - cS_i(t) + \\
+ &crr(t)P(t)(y_i(t) - y_i(t - \tau(t))) \text{diag}\left(\frac{k_m}{m + \Delta m}, \frac{k_m}{m + \Delta m}, \frac{k_I}{I + \Delta I}\right)\frac{d}{dz}\text{sat}(\sigma(t))\frac{dz}{ds}\bigg|_{\sigma(t)}
\end{align*}
\] (7)
Consider the behavior of the Lyapunov vector-function $V$ along the solution $(y(t), z(t))$ of the system (7) with initial function $(y(s), z(s))$, $-2\tau_0 \leq s \leq 0$, such that
\[
\max_{-2\tau_0 \leq s \leq 0} V'(y(s), z(s)) < \delta_1, \quad \max_{-2\tau_0 \leq s \leq 0} V'(y_3(s), z_3(s)) < \delta_0
\]

We estimate the derivative of the function (5) by virtue of (7) on the segments of the integral curves belonging in each point of the phase space to the sets
\[
\Omega_1[V_1] = \{ \varphi : [-2\tau_0, 0] \to R^4 : V_1(\varphi(s)) \leq LV_1(\varphi(0)), -2\tau_0 \leq s \leq 0 \}
\]
\[
\Omega_2[V_2] = \{ \varphi : [-2\tau_0, 0] \to R^4 : V_2(\varphi(s)) \leq LV_2(\varphi(0)), -2\tau_0 \leq s \leq 0 \}
\]

Here the constant $L = \text{const} > 1$ is chosen so that at all $i = 1, 2$ for all $t \geq 0$ the following inequalities are performed
\[
q_i(t, \gamma_1, \gamma_0, 1, 1, \gamma_0) + \frac{ck_3\tau_0 m \cos(2(1+c)a^2h_\tau_0')}{\gamma(m-\Delta m_0)} \max_{s \geq 0} q_i(s, L\gamma_1, L\gamma_0', 1, -1, L\gamma_0) \leq -\varepsilon_2
\]
\[
q_i(t, \varepsilon, \varepsilon, 0, \varepsilon, 0) + \frac{ck_3\tau_0 m \max_{s \geq 0} q_i(s, L\varepsilon, 0, L\varepsilon, 0, 0)}{\gamma(m-\Delta m_0)} \leq -\varepsilon_2
\]

Here $\varepsilon_2 = \text{const} > 0$. Is easy to see that under the conditions of Theorem 2 and 3 one can find this constant $L$.

Using the conditions 1 of Theorem one can be established that there is a point in time $t_i > 0$ such that $V_1(y(t), z(t)) < \gamma_1$ and $V_2(y_3(t), z_3(t)) < \gamma_0$ for all $t > t_i$. Indeed, the functional $V_1(t, y, z)$ on the set of segments $(y, z)$ of the integral curves of the system (7) belonging to the set $\Omega_1[V_1]$ for all $t \geq 0$ such that $\gamma_1 \leq V_1(y(t), z(t)) < \delta_1$, satisfies
\[
V_1(t, y, z) \leq \max \left\{ -\frac{h(c-1)}{mc} V_1(q_1(t, V_1, V_2, 1, 1, V_2), q_2(t, V_1, V_2, 1, 1, V_2) \right\}
\]
(8)

The functional $V_2(t, y_3, z_3)$ on the set of segments $(y_3, z_3)$ of the integral curves of the system (7) belonging to the set $\Omega_2[V_2]$ for all $t \geq 0$ such that $\gamma_0 \leq V_2(y_3(t), z_3(t)) < \delta_0$, satisfies
\[
V_2(t, y_3, z_3) \leq \max \left\{ -\frac{2a^2h(c-1)}{Ic} V_2, \frac{2(1+c)a^2h_\Delta I}{I(I+\Delta I)} V_2 - \frac{ck_3I}{I+\Delta I} \right\}
\]
(9)

Using (8) and (9) one can obtain the following estimations
\[
V_1(t, y, z) \leq \max \left\{ -\frac{h(c-1)}{mc} V_1(q_1(t, \delta_1, \delta_0, 1, 1, \delta_0), q_2(t, \delta_1, \delta_0, 1, 1, \delta_0) \right\}
\]
\[
V_2(t, y_3, z_3) \leq \max \left\{ -\frac{2a^2h(c-1)}{Ic} V_2, \frac{2(1+c)a^2h_\Delta I}{I(I+\Delta I)} \delta_0 - \frac{ck_3I}{I+\Delta I} \right\}
\]
Hence, we find that the time moment $t_1 > 0$ exists.

Similarly, using the conditions 2 of Theorem we can prove that there is a time point $t_2 > t_1$ such that $V_1^*(y(t), z(t)) < \gamma$ and $V_2^*(y_3(t), z_3(t)) < \gamma$ for all $t > t_2$.

Using the condition 3 of Theorem one can obtain that there is a time point $t^* > t_2$ such that $V_1^*(y(t), z(t)) < \varepsilon$ and $V_2^*(y_3(t), z_3(t)) < \varepsilon$ for all $t > t^*$.

### 4 SIMULATION RESULTS

The found control law has been applied in a numerical simulation of the robot motion. The calculations were performed for the following parameters

$$m = 20 \text{kg}, m_I = 3 \text{kg}, I = 3.76 \text{kgm}^2, a = 0.2 \text{m}, \Delta m = 1 \text{kg}, \Delta I = 0.1 \text{kgm}^2, h = 1.6 \text{Nc/m}$$

Parameters of the tracking trajectory are selected as:

$$\xi_0(t) = 0.2t \text{m}, \quad \eta_0(t) = 0.5 \sin(0.2t) \text{m}, \quad \psi_0(t) = t + 1 \text{rad}$$

The following control parameters are found

$$k_1 = 1, \quad k_2 = 0.5, \quad c = 1.1, \quad \gamma = 0.3$$

at the maximum delay $\tau_0 = 0.1 \text{c}$ and the area of initial deviations with $\delta_1 = 2$ and $\delta_0 = 0.5$.

The error tracking for the coordinates $\xi$ and $\eta$ is $\varepsilon = 0.05$. For the coordinate $\psi$ the tracking error tends to zero with increasing time. The integration time is $t = 50 \text{c}$. The trajectory of the robot (solid line) and track the trajectory (dashed line) are shown on the figure 2.

![Figure 2. The results of simulation.](image)

### 5 CONCLUSIONS

In this paper, the continuous tracking control problem for mobile robot with three omni-wheels is investigated. By considering the dynamic model of the system with parameter uncertainties and delay in the feedback, the Lyapunov functions method is used. Simulation results show the effectiveness of the method.
6 ACKNOWLEDGEMENT

The work was financially supported by Russian Foundation for Basic Research (15-01-08482).

REFERENCES


Interests and complementarities of two structural modeling tools for the analysis of multibody systems: MapleSim and Bond graph with 20-sim

BOUDON Benjamin*, MALBURET François

* LSIS laboratory (CNRS)
Arts et Metiers ParisTech
2 Cours des Arts et Metiers
13617 Aix-en-Provence – France
benjamin.boudon@free.fr

ABSTRACT

In the mechanical industry such as cars and helicopters, the difficulty of managing complex mechanical (or mechatronic) systems has led to a new category of tools called structural modeling tools. These tools have essential features that permit an efficient modeling of these complex mechanical systems: modularity, multiphysics and acausality. The main objective of this paper is to illustrate the advantages of structural modeling tools for the modeling of complex multibody systems and to analyze the complementarities between two structural modeling tools: Maplesim and Bond graphs conducted with 20-sim.

Keywords: Multibody systems, Functional models, Structural models, Modelica, MapleSim, Bond graph, 20-sim

1 INTRODUCTION

In the industry which designs complex mechanical (or mechatronic) systems such as cars and helicopters, to avoid certain undesired complex dynamic phenomenon, local and punctual curative solutions are often used. These solutions are often used because subsystems are modeled with modeling assumptions that are too strong for both their environment and structure. Additionally, they are built as an assembly of mathematical functional models and are not homogenous in terms of modeling tools. Moreover, these solutions are generally used at a specific moment without any generic approach and often for a specific version of apparel.

In this context, new structural modeling tools have emerged to tackle these difficulties. The main objectives of these tools are to bring a more global and modular modeling approach. A more global comprehension of physical phenomenon should lead to more sustainable solutions. A modular approach based on the oriented-object and acausal features will allow for a better knowledge capitalization. These structural modeling tools contain essential features that permit an efficient modeling of complex mechanical systems: modularity, multiphysics and acausality. In [1], the frame of the complex system in the aeronautic field is specified and a general presentation of some structural modeling tools have been presented.
The main objective of this paper is to illustrate the advantages of structural modeling tools for the modeling of complex multibody systems and to analyze the complementarities between two structural modeling tools: MapleSim and Bond graph with 20-sim.

For that purpose, two structural modeling tools: the bond graph (an energetic modeling tool) with 20-sim software (a simulation package for dynamic systems using iconic diagrams, block diagrams, bond graphs) and MapleSim are applied to a helicopter’s subsystem: the main gear box MGB-Fuselage joint. As far as MapleSim is concerned, this kind of modeling and simulation platform corresponds to a new trend in structural modeling tools coupling the benefit of Modelica models and symbolic manipulations [2].

First, the benefits of the structural tools in comparison with classic tools will be presented. Secondly, the main steps of the modeling of complex multibody systems with both MapleSim and Bond graphs with 20-sim will be presented. Thirdly, this paper will discuss the main differences between these two structural modeling tools and in what ways these tools are complementarity.

2 DEFINITION AND INTEREST OF STRUCTURAL MODELING TOOLS

An overview of the main families of modeling tools is first presented in this section. This presentation is of course not exhaustive given the large number of existing methods and modeling tools. It is necessary to specify that these tools are centered on the pre-design phase which is the phase studied mainly in the frame of the dynamic of multibody systems. A first proposition of the classification of these main modeling tools has been proposed in [1]. The main feature of this classification is to separate the functional tools from the structural ones. There are no official definitions of the functional and structural modeling tools. However, inspired from [3], definitions are proposed in the next section. This section aims to describe the specifics of each of these tools.

2.1 Functional modeling tool

Definition: The functional models are defined as an assembly of mathematical functions where each function is defined by one or several inputs and one or several outputs.

These models represent how steps for calculations are performed. However, with these tools, the system architecture no longer appears explicitly. In others words, in the scope of multibody systems, the position between the elements (bodies, joints, springs...) is not accessible without detailed analysis of the mathematical functions. The main functional modeling tools will now be described. They correspond with the first tools used for conducting simulations in Engineering.
2.1.1 Equations

The oldest modeling technique is that which consists of translating the behavior of a system into a mathematical model in the form of equations to be solved. The resolution can then be performed analytically or numerically. An analytical resolution is performed when explicit formulas to obtain the solution are possible. When possible, the analytical methods are preferred because they are accurate and allow the analysis of the influence of various parameters on the result. The numerical resolution is used when there is no possible analytical development or when the number of equations is too big [3]. It allows using a set of algorithms to obtain a numerical solution to a problem with a desired accuracy after a number of arithmetic operations. Therefore it is generally sought to avoid or delay the last possible numerical solving. When only a numerical solution is possible, it is interesting to preserve an analytical calculation phase before the numerical solving [4].

Currently, the traditional tools used for handling and solving analytical equations are called formal or symbolic software. In [5] a number of advantages of symbolic computation are mentioned. For examples, one may cite Maple or Mathematica, MuPad. For the numerical resolution, much software exist. For example, one may cite programming languages like C++ or software with completely digital scientific calculations such as Matlab or Scilab.

2.1.2 Block diagrams

Block diagrams modeling have been known since the beginning of automatic. Also, only their essential characteristics will therefore be recalled. The block diagrams represent functional relationships between inputs and outputs. Modeling with block diagrams is said to be informational (or signal type) to the extent that the links between the blocks provide only information and no energy transfers as with bond graph, which will be described in following this chapter. Functional models can be hierarchical [6] but the decomposition of these models is then carried in a functional manner, that is to say that a function can be decomposed into sub-functions.

2.2 Structural modeling tools

In the last twenty years, new modeling tools called "structural" compared to older and more classical tools called "functional" have appeared. As for the functional modeling tools, this new type of tool is characterized by their production of structural models.

**Definition:** The structural models are defined as an assembly of models of physical elements or subsystems following the same architecture as the real system without prior analysis on how to connect the various subsystems.

The structural modeling consists of determining the elements and structure (relationships between the elements) of the system to achieve the technical functions of the system. To better understand this concept, structural models can be superimposed on physical schemas (the kinematic schemes for multibody systems) used at the beginning of the modeling phase so as to represent, in a simplified way, the topology of the system.

2.2.1 Specific tools

They are monophysics structural tools. For example, we can mention ADAMS, LMS Virtual Lab or SIMPACK. These tools are very powerful in their fields and can now be interfaced with functional tools such as Simulink to conduct control/command simulations. Nevertheless, they don’t allow the direct modeling of components from other physical domains and require co-simulation for the simulation of multiphysics system [7].
2.2.2 Modelica

Modelica is an object-oriented modeling language intended mainly for multiphysics modeling systems. It was created shortly before 2000 and has been developed by the non-profit association Modelica based in Sweden. In this paragraph, only the essential properties of language will be discussed.

An object is a concept that was created to bring together in a single entity a set of procedures on the same data set. In the Modelica modeling, all the elements used to build the models are defined by a set of classes. A class is a family of objects that share common properties. In [8], the definition of the seven classes used can be found. An instance is an object of a class and therefore includes all properties of the class from which it came.

The main properties that seem particularly useful are: its characteristics as an object-oriented language, its acausal behavior, its multiphysics characteristic and its graphic layout. First, as the object language, Modelica language facilitates the creation of hierarchical models and reusable libraries (because they are very generic). In addition, the modification of a model or changing a parameter is also very easy. Also, the Modelica language is a language allowing the establishment of acausal models for which the computation causality at the equations level linking the variables and parameters is not assigned until the code is not compiled. In addition, this property enables the models to be more reusable since different types of computations can be conducted with the same model. Finally, the Modelica language is used to describe a unique language of different areas of physics.

Thus, Modelica seems to gather the necessary features of modularity (hierarchical, scalable, reusable) for the modeling of complex mechatronic systems in a structural way. In addition, the models are easy to access for a neophyte as specified in [8].

2.2.3 Symbolic tools such as MapleSim

In addition to the Modelica features that have been mentioned, these kinds of tools, such as MapleSim, have a symbolic computation engine. The benefits of a symbolic computation are present in the modeling phase until the simulation phase. First, the symbolic computation allows the analysis of the selection of coordinates on the complexity of equations. Next, a symbolic analysis can be made from the equations of the dynamics (calculation of jacobian matrix, determination of closed form solutions). Finally, in regards to the simulation, symbolic techniques for the index reduction of DAEs can be conducted: deduction of the set of minimal coordinates, elimination of redundant systems.

2.2.4 Bond graph

The concept of energy is fundamental in the description of the evolution of technological systems. Energy is present in all areas of physics and is the link between them. From this observation, a number of tools with energetic representations for modeling complex systems have been defined. One of the main tools is the bond graph (BG). The bond graph was created by H. Paynter [9] in 1959 and developed by R. Rosenberg and D. Karnopp [10] at MIT Boston in the United States.

The bond graph is based on a study of the transfer of power in a system modeled by lumped parameters. The bond graph is a graphical modeling tool that covers all physical systems (mechanical, hydraulic, electronic, thermal...) regardless of their condition (linear, nonlinear, continuous …). It is represented as an oriented graph showing dynamic variables and power bonds between these variables. The bond graph systematically associates two different variables for each bond: a generalized effort variable (which is a force or a torque in mechanics) and a generalized flow variable (which is a translational or rotational velocity in mechanics) on each
side of the half-arrow link. Each bond has therefore power information, obtained by the product of these two variables, and allows direct access to the energy transferred by a simple integration of power. The bond graph approach enables thus, the representation of mechatronic systems in a graphical form that describes the exchange of power between basic elements like inertia, compliance, dissipation, conservative power transformation, gyrator actions and sources. More details on bond graphs can be found in [11], [6] details the possible construction and operation.

The main benefits of bond graphs are presented in the following.

On the one hand, it facilitates a systemic approach necessary to design a mechatronics system thanks to its features of object-oriented language and acausality (in a similar manner to modelica).

First, the multilevel representation of the system permitted thanks to the use of the word bond graph (WBG) allows for the concatenation of the bond graphs’ bodies and joints. This technique makes it possible to "zoom in / out" on different parts of the system, such as in a Simulink model. Secondly, the modularity allowed by the MBG method enables the model to evolve to meet the levels of complexity required for each design problem by the addition or modification of new components and subsystems and by replacing behaviour laws. As a consequence, the global representation of the system built from subsystems facilitates the management of interactions and/or couplings. Thirdly, the structural approach generally enables the generation of an acausal model, which makes its structure independent of its inputs and outputs.

On the other hand, bond graphs also enable the use of new features such as energetic analysis and structural analysis of the system from the bond graph structure (causality, controllability, observability, inversibility, etc). These characteristics will be further developed in section 4.2.

3 MODELING AND SIMULATION WITH STRUCTURAL MODELING TOOLS

3.1 Methodology of modeling

As it was defined previously, structural models are built as an assembly of models of physical components or subsystems.

3.1.1 With MapleSIM

Regarding the building of models

To build a multibody system with MapleSim, components (bodies, joints, mechanical actions, …) are dragged from the 'Library Browser’ into the modeling window. Contrary to modelica, extra coordinate frames are constructed by adding “Rigid Body Frames”. These blocks can generally be used to define intermediate frames with respect to the inertial frame in a rigid body model.

By clicking a component, the 'Parameters’ window (as you can see in Figure 3) displays the possible parameters that are applicable for the component that is concerned. For a rigid body, mass properties can be specified as well as some initial positions and orientation properties. This interface allows for the entering and modifying of input data into the system. MapleSim permits a hierarchic view of the model that enables the easily identification of different levels of subsystem and components. As can be seen in Figure 3, MapleSim uses different colours and symbols for connection-lines and different connection-ports to make a distinction between different sorts of signals such as multibody-signals or normal signal lines and between inputs or outputs.
Simulating a model

Due to the constraints equations that come from kinematic loops or the choice of absolute coordinates, the equations of mechanical systems can be differential algebraic equations (DAEs) whose resolution requires specific numerical integration methods. A recent review of the methods for solving DAEs can be found in [12]. Mainly depending on the nature of the constraint equations, the index of the DAE enables the qualification of the complexity of the DAE problem. MapleSim uses a combined approach of symbolic reformulation and a specific DAE solver that solves DAEs with a high-index.

Simulation results

Results can be plotted by attaching probes to the connection lines of the model. After attaching probes and simulating, a 'Plot Window' appears. Graphs can be exported to various types of image-formats or data with .csv or .xls file.

3.1.2 With BG conducted with 20-sim

The building of models

Components (bodies, joints, springs, dampers,...) used in multibody systems are not furnished by the libraries of 20-sim. Consequently, for the modeling of 3D multibody systems, the user has to build his own library. A detailed review specifying applications of BG modeling for MBS can be found in [6]. In order to keep a modular approach, the principal method for modeling multibody systems with bond graphs is the BOS and TIERNEGO method [13]. This method enables a multibody system to be built as an assembly of bodies and joints. The principle of this method is based on the use of absolute coordinate systems and Newton equations. Indeed, in such a way, the dynamic equations of a rigid body depends only on its own geometric parameters and mass/inertias parameters and consequently, the dynamic equations of the complete system consists of a sum of the equations of each body.

Even if bond graphs still have important specific features in comparison to other structural modeling tools, these latter (LMS Virtual Lab, Modelica, MapleSim, SystemModeler ...) present the advantage of having a user-friendly interface for entering data. Moreover, this practical aspect: a lack of user friendliness (in comparison to software dedicated to multibody system modeling) may have a negative consequence since the modeling task can be longer and therefore more prone to mistakes. From another point of view, contrary to software dedicated to multibody modeling where the multibody elements have finite possibilities of parametering, the MBG is more completely configurable since the designer builds the multibond graph of bodies and joint built from the standard elements depicting physic laws and which can be thus easily modified.

Moreover, the use of bond graphs for multibody systems requires a certain level of expertise. This point alone makes the system less attractive to many engineers since the formation of a tool can become too time-consuming. Indeed, in order to create its first multibody model, a mechanical engineer needs to learn classical bond graph modeling [10, 11, 14]. Next, he has to analyze how to model multibody systems [15-17]. Moreover, as we mentioned before, there is no official procedure and design environment dedicated to the modeling of multibody systems. Consequently, sharing models requires some cautious.
Simulating a model

Due to the constraints of equations that come from kinematic loops or the choice of absolute coordinates, derivative causality appears on the inertial elements in the bond graph models and leads to differential-algebraic equations. As described in [18], numerous solving methods provided by the Multibody community exist for the simulation of multibody systems with 20-sim. For example, we can cite some of them. First, DAEs can be transformed into ODEs by different methods such as singular perturbation (also called virtual springs) or minimal coordinates. Secondly, the “Lagrange multiplier” method on the kinematic constraints associated with a constraint stabilization method such as Baumgarte can be conducted. Thirdly, thanks to variable step variable order DAE-solver based on DASSL, DAEs can also be solved directly when the index reduction is inferior or equal to 1.

Simulation results

With a similar method as the probes in MapleSim, results can be plotted in the simulation environment by selecting the energy or signal quantities. In contrast to the probes in MapleSim, energy or signals can be visualized during the simulation in 20-sim. Also, graphs can be exported to various types of image-formats or data with .csv or .xls file. The data exportation allows for the comparison the results from different sources (for example MapleSim and 20-sim) in the same graphic interface (for example Matlab).

3.2 Example of the MGB-Fuselage model

3.2.1 Context

In the frame of the “Complex mechanical system dynamics” project (an European Aernautic Defense and Space foundation (EADS)), research has been conducted on the modeling of helicopters subsystems with an energetic tool: bond graph. MapleSim has been used for the verification of the BG model.

The case study presented here is a specific suspension which filters the vibration coming from the rotor to the fuselage. This subsystem is: the Dynamic Anti-Resonant Vibration Absorber system (DAVI) (called Suspension Antivibratoire à Résonateur Intégré (SARIB) in French). The SARIB system is a complex mechanical system with closed kinematic chains (CKC).

Briefly, the SARIB system is composed of SARIB bars with a tuning mass on each bar that is installed between the MGB bars and the fuselage. The SARIB system is designed so that the tuning mass creates inertial forces opposite to the force of the MGB bars on the beaters. This system reduces the efforts transmitted from the rotor to the fuselage for a frequency called anti-resonance frequency. More details can be found in [1], [19].

3.2.2 Kinematic scheme

A kinematic scheme of the complete MGB-Fuselage is shown in Figure 2. The system consists of ten rigid bodies: a main gear box (MGB), a fuselage, four MGB bars, and four SARIB bars (or beaters). Each leg consists of a SARIB Bar and a MGB Bar connected by a spherical joint. The upper ends of these legs are connected to the MGB with spherical joints and the lower ends of these legs are connected to the fuselage through revolute joints. The central membrane located between the MGB and the fuselage is modeled with a prismatic joint and two revolute joints in serial.
3.2.3 Models

The general aspect of the MapleSim models and bond graphs of the MGB-Fuselage joint are presented below. More details about the construction of the bond graph model in this system can be found in [1].

Figure 2 – 3D Kinematic scheme of the anti-vibratory subsystem (DAVI)

Figure 3 – MapleSim model of the MGB-Fuselage joint
3.2.4 Results

As mentioned before, both MapleSim or 20-sim enable the modeller to select the variables needed to measure. MapleSim has permitted to check the BG model built with 20-sim. The main simulation parameters used were the following:

<table>
<thead>
<tr>
<th>Table 1. Simulation comparison</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Software</strong></td>
</tr>
<tr>
<td><strong>Solver</strong></td>
</tr>
<tr>
<td><strong>Methods</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td><strong>Number of coordinates</strong></td>
</tr>
<tr>
<td><strong>Number of constraints</strong></td>
</tr>
<tr>
<td><strong>DoF</strong></td>
</tr>
</tbody>
</table>

So, to analyse the frequency behaviour of the system, the MGB was excited with a chirp signal and the vertical acceleration of the center point of the fuselage has been measured. As expected, an anti-resonance phenomenon is present around the habitual excitation frequency of the rotor.
The slight difference between the two curves is from the use of complementary springs and dampers (even if they are very stiff) in the singular perturbation method for the bond graph model.

4 DISCUSSION ABOUT THE COMPLEMENTARIES BETWEEN MAPLESIM AND BG CONDUCTED WITH 20-SIM

4.1 Benefits of the MapleSIM

4.1.1 Quick modeling

MapleSim enables the modeler to obtain an easy modeling thanks to its library of components and the interfaces that allow the entering of all parameters in a systematic way. As such, MapleSim can be very useful in the verification phase. Moreover, many troubleshooting tools, such as the possibility to hide some subsystems, allow for a quick convergence to a working model.

4.1.2 Symbolic computation

MapleSim doesn’t only enable simulations to be conducted. MapleSim also enables the modeler to conduct symbolic analysis. For the exploitation of a BG model, the reduction of equations may be necessary since some techniques have only been given for ODE systems. One way to reduce the equations is to transform the coordinate system into a minimal set of relative coordinates [20] and to express the dependant coordinates as a function of the independent coordinates. Thus, MapleSim can be particular useful. Indeed, during the modeling phase, different types of coordinates can be chosen (absolute coordinates, joint coordinates, hybrid coordinates) and the transformations from one set of coordinates to another set of coordinates can be done automatically. Moreover, in a model with relative coordinates, the MapleSim’s symbolic computational tool determines a closed form solution of the dependant coordinates as a function of the independent coordinates. The principle of this approach has been described in [2]. By following this approach, a minimal set of angular coordinates which enable the positing of a branch MGB bar-Sarib Bar have been expressed as a function of the set of angular coordinates positioning the MGB with regard to the fuselage. To say it in robotic terms, the inverse kinematic problems have been solved symbolically and in an automatic manner.

4.2 Benefits of BG models conducted with 20-sim

Many exploitations of bond graphs models can be carried out on different tasks (simulation, sizing, command, diagnostic, energy analysis …) [6]. Besides, some of these properties are specific to bond graphs (linked to intrinsic properties of bond graph: energetic bonds and causalities) and they are an important added value of the bond graphs compared to other modeling tools. The tasks of exploitation which can be considered the most valuable for the
multibody research community are: the structural analysis of bond graphs, the control-command determination due to the inversion of BG models and the energetic analysis. In this section, these bond graph exploitations will be enounced without any results since their applications on the described system are not enough mature enough to be presented.

4.2.1 Energetic structure

The analysis of the energetic exchange in a mechanical structure can be particularly interesting in determining the energy sinks in a structure, that is to say the places where some important non-used energy can be exploited. For example, this is typically the case for helicopters where the high level of mechanical vibrations could be used for vibration energy harvesting. Depending on the volume of energy scavenged, the interest of energy harvesting helps make some electronic devices autonomous. For example, autonomous sensors permit the removal of expensive and bulky wires or batteries that can be problematic in terms of weight, lifespan and maintenance.

4.2.2 Structural analysis and model inversion

The structural analysis of the bond graph [21] is based on the properties, which are dependent only on the type of bond graph element in the system, the way they are connected, independently of the numerical values of the parameters of these elements. Due to the structural analysis, decisional information for the controller designer such as controllability, observability, and invertibility, which can be derived directly from the structure of the bond graphs, can be obtained.

The determination of a control-command can be conducted by the inversion of the bond graph models. The model inversion is based on the principle of solving the input of systems in terms of the output and states. Thus, with this inversion model method, the command inputs can be determined straightforwardly from the specifications of the outputs. In other words, this method allows for the structure of control law directly from the bond graph with less additional tunings (to say it, in a more simplified manner, in theory, the control approach can almost be used in open loops) than in the classic closed loop scheme based on the classical approach “try-error-correction”.

5 CONCLUSIONS

In this paper, the main features of structural modeling tools have been presented and compared to the functional tools. Next, the methodology for building structural models has been developed for two tools: MapleSim and bond graphs conducted with 20-sim. The example of the MGB-Fuselage system was used to illustrate the type of models and an example of the results obtained was also presented. Some benefits of using MapleSim in a complementary way with the study which has been done on bond graphs have been given. MapleSim has permitted a quick verification of the BG model and to conduct symbolic manipulations so as to reduce the system of equations underlying the BG model. Some possible benefits of bond graphs (energetic analysis, structural analysis, model inversion in comparison to other structural modeling tools such as Modelica, MapleSim have been mentioned. However, as results are still not mature, they will be exploited in future works.
REFERENCES


ABSTRACT

The availability of various multi-body dynamics software (MBDS) always motivates one to verify and compare in-house dynamics algorithms for analyses of complex mechanical systems. For instance, studying a 6-degrees-of-freedom (DOF) Stewart Gough platform and comparison of its kinematics and dynamics results with in-house developed algorithms would require building a correct MBDS model first. Very often, a mismatch between the results of MBDS and the in-house algorithms misleads a researcher. In many cases, such effects are due to the improper methods of modelling and comparative analysis, rather than a common error in the in-house dynamics algorithm. This paper focuses on issues in building, analyzing, and comparing the spatial manipulator models built in a MBDS with the in-house developed algorithms. To illustrate the issues, a CAD model of the Stewart platform was built using RecurDyn MBDS and its kinematics and dynamics analysis results were compared with the in-house model in MATLAB. The issues addressed and the guidelines provided in this work would certainly reduce the developmental and analyses time for spatial systems in a MBDS.

Keywords: Multi-body dynamics software, Stewart platform, CAD models.

1 INTRODUCTION

Spatial manipulators are of special interest in the robotics community due to their inherent nature of spanning a three-dimensional physical workspace. Specifically, spatial manipulators with closed-loop topology are largely preferred over their open-loop counterpart, particularly in situations where precise positioning, large payload capacity and higher stiffness of the system is desired. Several manipulators of such configurations have been proposed in the literature viz. 3RPR manipulator [1], 3-DOF delta manipulator [2], and 6-DOF Stewart platform [3]. A few popular ones used commercially as a robot and/or haptic device is the 3-DOF delta manipulator [4].

Most often one is interested in developing new modelling strategies and efficient simulation algorithms for analyses of such systems. For validation of kinematic analyses and the dynamic model of a particular system, one often makes use of commercial multi-body dynamic software, viz. ANSYS [5], ADAMS [6], Autodesk Inventor [7], RecurDyn [8], SimMechanics [9], etc. However, in many instances, particularly for the spatial manipulators, the modelling and analyses part takes quite large time. This large amount of time is generally spent in validation or comparison of the results of the commercial MBDS with the in-house developed algorithm. Very often the time spend on modelling such systems is lesser compared to the validation/analysis part.

In this paper, we discuss the issues that we faced while modelling a 6-DOF Stewart platform in RecurDyn, or more general in any MBDS. In particular, we highlight the issues that we faced while comparing the results from a MBDS model and the in-house algorithm. The issues addressed and the solutions/methods presented would greatly help a researcher in saving considerable time for analyses of such systems with a MBDS. The rest of the paper is organized as follows: Section 2 will discuss the modelling steps in RecurDyn and the key issues there.
Section 3 and 4 will discuss the issues in kinematic and dynamic analyses of spatial manipulators, respectively. Comparison issues are detailed in Section 5, followed by Conclusions in Section 6.

2 MODELLING A SPATIAL MANIPULATOR IN RECURDYNE

In this section we discuss the steps carried out for modelling a spatial manipulator in RecurDyn. Recursive dynamics (RecurDyn) from Function bay Inc. is a popular dynamics analyses software for modelling and simulation of both rigid and flexible multi-body systems. The software has versatile capacity for modelling standard to arbitrary shapes as a component of any multi-body system. We considered a 6-DOF Stewart Gough platform as an illustration or case study in this paper. The manipulator is popularly used in flight simulators, vibration isolation, robotic surgery, etc.

Figure 1 depicts an in-house developed 6-DOF Stewart platform consisting of six symmetrical legs with two similar irregular hexagons acting as a fixed base and a moving platform, respectively. It was developed as a part of M-Tech/Ph.D. research at IIT Delhi to validate some of the kinematics/dynamics algorithms for trajectory planning, singularity avoidance/passing, etc. The manipulator can achieve three translations and three rotations about the corresponding Cartesian axes, respectively. Any spatial motion of the platform can be achieved by actuating six symmetric legs of the manipulator, which are driven by electric motors/hydraulic/pneumatic drives.

An optimised kinematic model was chosen to achieve a suitable non-singular workspace for the system in Fig. 1. Figure 2 depicts the corresponding RecurDyn model. Modelling in RecurDyn consisted of two major steps, namely, 1) Modelling 6 symmetrical legs, and 2) Modelling fixed and the moving platform.

2.1 Modelling of the platform

The manipulator in Fig. 1, had two platforms. One of them being fixed with the ground, representing the base of the manipulator, while the other being floating, representing the moving platform or the end-effector of the manipulator. The shape of the platform in Fig. 1 is typically an irregular hexagon, which is modelled using ‘general’ command under the category ‘body’ of RecurDyn.

The dimensions of these irregular hexagons are derived from an equilateral triangle, whose corners are assumed to be chipped off to achieve a six sided platform, whose three opposite sides are equal in length. Extrusion facility in the general modelling environment enables one to model a solid object out of the two-dimensional drawing. After choosing a particular material, the inertial properties and the volume of the platform are automatically available in the properties dialog box.
2.2 Modelling of manipulator legs

The legs of the manipulator are composed of two parts, which achieve relative sliding motion on account of a prismatic joint between the two. In the actual manipulator, the legs are connected via spherical joints with the fixed and moving platform, respectively.

For simplicity of modelling in RecurDyn, legs were assumed to have cylindrical shape. Two cylinders of different diameters were used to model the two parts, whose lengths were chosen based on the dimensions of the actual manipulator. Since the initial orientation of the legs with the base was unknown, the cylinders were modelled on the line joining the base and the corresponding platform corner. This step proved out to be tricky, as the design of cylinders in the MBDS requires parameters, namely, the location of centres of the two circular ends and the radius of the cylinder. In particular, RecurDyn has limitations for defining such lengths directly. Hence, an STL file of the cylinder was generated, and later imported to Autodesk Inventor environment. The length of the cylinder was easily changed in the Inventor environment using the modify dimension option. The link was exported back to RecurDyn, maintaining the orientation desired.

---

For proper kinematic definition, one prismatic and two spherical joints were defined, connecting the two parts of the leg and the same to the fixed and moving platform, respectively. Figure 3 depicts the platforms with a single leg attached by joints. Similarly, five more legs of the same dimension were attached to the corresponding positions on the fixed and moving platform. In this way the complete model of the 6-DOF Stewart platform was developed, as shown in Fig. 2.

3 KINEMATIC ANALYSIS

Typically in parallel manipulators, the trajectory of the platform is of interest. Any non-singular trajectory of the platform can be achieved by proper leg actuation. This leg actuation is given in a MBDS by defining a suitable joint between the moving platform and ground. However, dynamic analyses using such technique would not estimate the required forces at the actuators in a typical MBDS. Hence, inverse kinematic analysis is of concern, where the platform is moved at desired trajectory, and the position, velocity and accelerations of the joints are determined. Joint variables are required as the pre-processing component for the dynamics analysis that will be detailed in Section 4.

In any MBDS, in order to move a body, an input motion needs to be given by defining suitable joints connecting the body. It is to note that while defining the joints the selection of sequence of the bodies is important. For example if the joint is between the ground and any other body, the ground should be selected first for standardization. Similar is the case when the joint is to be
applied between the two relative bodies, as in the case of the prismatic joint in the leg assembly of the Stewart platform.

The Stewart platform being a 6-DOF motion platform, can handle six independent inputs namely, three rotational (roll, yaw, pitch) and three translational (heave, surge, and sway), as shown in Fig. 2. The facility to impart all six motions at a time is not generally available in a MBDS. At a time, one can only impart a maximum of two motions along a particular axis only by selecting some 2-DOF joints, e.g., cylindrical joint, prismatic joint, etc. If it is required to move the platform with a specific trajectory a special motion under PTCV available in the professional toolbox can be used. For PTCV a curve is to be defined, which will become the trajectory of the platform. The joint variables for the prescribed platform motion can then be found using the plot command in the analysis tool bar of RecurDyn. It is to note that motion trajectory can be given as time varying equation or a set of points, and it should be ensured that the trajectory should be non-singular.

4 DYNAMICS ANALYSIS

A general spatial motion to the Stewart platform end-effector can be imparted by proper actuation of six symmetric legs. Here the problem of dynamics analysis is to find the forces required to obtain desired trajectory (position, velocity and acceleration) of the platform. Each leg assembly comprises of a prismatic joint. In order to know the forces at these joints, the motion should be given directly at the joint in a MBDS. This is one of the limitations in MBDS that it determines the forces/torques only at the joints where input motion is given.

To overcome this limitation alternatively, the joint motions are generated through inverse kinematic analysis as described in Section 3. These joint motions are then interpolated with a spline fit available in ‘SubEntity’ toolbar for each actuating prismatic joint separately. The data from spline fit is then used at the respective joints using the command AKISPL. Joint forces can now be obtained by performing the dynamics analysis in the MBDS. The step to execute this process is shown in Fig. 4, where the creation of spline and its use for translational joint using ‘include motion’ is also depicted.
5 COMPARISON OF RESULTS

For proper comparison of the results of the proposed algorithms with any MBDS, the kinematic and inertial properties of the system should be same. To ensure this, the effort starts with building of the identical model, and identical kinematic and dynamic parameters of MBDS model, which are the input for the proposed algorithm. Some key points to focus while building the model are discussed in Section 2. In the RecurDyn model, the initial height and orientation of the platform, the numbering of legs, etc. should be same to have the same kinematic structure.

In this example, the kinematic parameters are the joint variables. The dynamic parameters include mass, inertia tensor, mass centre location. The dynamic behaviour of the system is highly influenced by the mass and inertia properties of the bodies and the coordinate frames in which they are referred.

The Stewart platform comprises of six parallel legs assembly and the moving platform. The leg assembly includes upper and lower leg coupled with prismatic joint. Dynamic properties of each body should be transferred correctly. There is no issue in taking the mass, but while taking the values of inertia tensor from the properties dialog box of MBDS, care needs to be taken to check that the pose of the body fixed frame of both the in-house algorithm and MBDS should be identical. If it is not similar, then the respective rotation matrix should be used to match the two configurations. Alternatively, one can draw the complete system separately, as per the Denavit Hartenberg (DH) frames of the proposed algorithms. The mass centre location is also to be checked, due to the reason that in Stewart platform, the origin of DH frames of the legs in proposed algorithm lies at the same point. Figure 5 shows the mass and inertia properties of the moving platform from the properties dialog box.

Taking into consideration all the issues and modelling hurdles, the results of the dynamics analyses were compared with the in-house algorithm using MATLAB [10]. Figures 6 and 7 depict the inverse dynamics results for the heave and yaw motion of the Stewart platform.
SUMMARY

Building a complex model in any MBDS and its comparison with in-house algorithms helps to understand the behaviour of real systems while simultaneously acting as a good debugging tool. Care has to be taken while modelling a system in any MBDS, which includes proper selection of the coordinate frames and similar link inertial properties. In this work, several issues in modelling of a spatial system in a MBDS were detailed. Besides several limitations of the current MBDS were reported in this part. The issues addressed and the solutions/methods presented in this work would thus greatly help in saving considerable time for analyses of spatial systems with a MBDS.

ACKNOWLEDGEMENT

The work presented in this paper was conducted at IIT Delhi as a part of Ph.D. research of first two authors. The partial assistance from SDD, Secunderabad and BARC, Mumbai, extended to the first and second author is sincerely acknowledged. The second author greatfully acknowledges Prof. S. K. Saha for supporting from his Naren Gupta Chair Professor fund for presenting the paper in Barcelona, Spain.

Figure 6. Inverse dynamics results for the yaw motion (links 1, 3, 5)

Figure 7. Inverse dynamics results for the heave motion
REFERENCES


Robotran-Yarp interface: 
a framework for real-time controller development 
based on multibody dynamics simulation

Timothée Habra∗, Houman Dallali#, Alberto Cardellino#, Lorenzo Natale#, Nikolaos Tsagarakis#, Paul Fisette∗, Renaud Ronsse∗

∗ Center for Research in Energy and Mechatronics
Université catholique de Louvain (UCL)
Place du Levant 2, 1348 Louvain-la-Neuve, Belgium
[timothee.habra, paul.fisette, renaud.ronsse]@uclouvain.be

# Advanced Robotics & iCub Facility
Instituto Italiano di Tecnologia (IIT)
Via Morego 30, 16163 Genova, Italy
[houman.dallali, alberto.cardellino, lorenzo.natale, nikos.tsagarakis]@iit.it

ABSTRACT

Multibody dynamics simulation is widely used for testing and prototyping controllers. However, the transfer of controllers initially developed in simulation to real mechatronics platforms, requires modifications of the code in order to interface with sensors and actuators. Because of this coupling with the hardware, the controller re-usability is severely impacted.

In this work, we solve this issue by adding a middleware between the controller and the controlled platform (real or simulated). This framework decouples the controller from the hardware which allows fast controller development and eases collaborations on large scale projects. Moreover, it is then possible to simultaneously control the real and the simulated robot from a unique controller. This paper presents the interface of the Robotran dynamic simulator with the YARP middleware. Robotran leverages the symbolic generation of the multibody equations to provide fast and accurate simulations of multibody systems. Thanks to the speed and accuracy of Robotran, it is possible to test real-time controllers in a realistic simulation environment.

This framework is illustrated with applications on the COMAN humanoids robot.

Keywords: Software development, Multibody simulation, Middleware, Robotran, YARP.

1 INTRODUCTION

Simulation tools are widely used in testing and prototyping new technologies. By providing a safe and controllable testing environment, they allow fast and cheap code prototyping. Benefits of dynamics simulation are particularly evident for robotics controllers development. Indeed, robotic platforms are not always available for controller testing (long mechanical design, hardware repair, few platforms) and running untested controllers turns out to be unsafe (operator injuries, hardware damage).

Nevertheless, transferring a controller developed in simulation to a real robot is not straightforward. It typically requires to adapt the format of the input and output signals of the controller to the robot actuators and sensors. Not only time consuming and error prone, this coupling of the high-level controller with the robot hardware severely impacts the re-usability and lifespan of the code. The code stays with a specific robotic platform and usually gets obsolete as soon as the hardware changes.

To tackle these problems, several middleware tools were developed within different robotics frameworks such as YARP [2], ROS [3] and Orocos [4], to name a few. By decoupling the controllers from the robot hardware, they encourage code reuse and collaboration across projects. A unique controller can work with different pieces of hardware (e.g. new joint encoders) as long as they are interfaced with the middleware. This approach allows to run the user controller with a simulator that can be seen as just another hardware, letting the middleware taking care of matching
the relevant input/output signals. Therefore, without changing a single line of code, the very same controller can work with the simulator and with the real robot.

This paper presents the coupling of the Robotran multibody dynamics simulator [1] with the YARP middleware [2]. The proposed framework allows fast controller development and eases collaborations on large scale projects. Robotran was selected as the dynamic simulator for its speed and accuracy. To illustrate how this framework can be used for controllers of complex mechatronic systems, an example with the COMAN humanoid robot is given. The corresponding code is open source1.

2 YARP MIDDLEWARE

A major problem in robotics is that controllers can quickly get entangled with the platform they are running on and the devices they are controlling. In that case, as soon as a piece of hardware (e.g. on-board PC, sensors) is changed, controllers become obsolete and need to be modified. This is particularly burdensome nowadays with advanced devices being constantly improved and updated, especially in research field. In that context, the YARP robotics framework has been developed to enhance code modularity and re-usability.

A key idea of YARP is to factor the hardware-controller interface out of the controller code. As shown in Figure 1, it is moved into a middleware that provides the interface for actuators and sensors of the robot. Thus, the controllers are totally independent of the hardware. People developing controllers can then focus on their algorithm and are set free from the interfacing work with the devices. The lifespan of the code is then extended as it does not need to be modified when some pieces of hardware are updated or changed.

![Figure 1](image)

**Figure 1.** Blue rectangles represent the piece of code specific to the hardware. A) Controllers directly commanding the robot, with their hardware-specific code. B) Controllers interfaced with a middleware have no hardware-specific code.

Practically, the controller receives sensor measurements as inputs and sends control commands as outputs through some *interfaces*. Interfaces consist of API (Application Programming Interface) 1

---

1 in public repositories https://gitlab.robotran.be/walkman
to control actuators and read sensors data. They are implemented in C++ as abstract base classes with a set of virtual methods. So for each type of sensors (e.g. joint position encoders), there are interfaces specifying the set of functions to access the data (e.g. \texttt{getEncoder()}). Similarly, for each type of controlled actuators (e.g. a joint DC motor), an interface specifies the functions to control it (e.g. \texttt{positionMove(ref)}). For more details about YARP interfaces, see [5].

For each actuator or sensor device interfaced with YARP, a \textit{driver} should implement the corresponding \textit{interface} classes. It is important to note that drivers of different devices belonging to the same family can implement the same interface. For example, different type of encoders may implement the \textit{encoder interface}. Thanks to this architecture, the controller is device agnostic: it knows the interface functions, but does not need to know which device implements them. All that matters is that the user knows the type and format of the data he/she gets or needs to send. So each device can implement interfaces when appropriate, with the requirement of being compliant with the specifications, e.g. regarding the unit (degrees for angular positions, Newton for forces). This ensures that the controller can work with any platform implementing its interfaces as illustrated in Figure 2. This modular approach is similar to a computer where you can change a device (screen, keyboards,... etc) with no need to change your programs.

In addition, for each interface, YARP implements generic network proxy devices that allow remote execution of the same code. Therefore, different controller pieces can run on different machines. Thus, controllers on a user laptop can be interchangeably connected to the robot or the simulator with no need to recompile. That allows to take advantage of specialized computer for specific computations (e.g. for vision or intensive computation).

3 ROBOTRAN-YARP DRIVERS

Since a simulator can be seen as a virtual robot, it is possible to implement drivers for the simulated devices. This was done for the Robotran simulator. It allows to test controllers in simulation and then to transfer them to the real robot without changing a single line of code. Figure 3 illustrates an example of this structure with the COMAN robot.

Similar interfaces between middleware and simulators already exist: for example YARP with

![Figure 2. Different devices of the same family, Type A Motor and Type B Motor are controlled by the same controller. Each motor has its own driver but both drivers implement the same interface.](image)
Gazebo [6] and ROS with Gazebo\(^2\). Gazebo simulator is attractive as it allows to model complex world and vision sensors. Moreover, Gazebo and ROS communities are large and active. However, none of the existing solutions was feeling our needs. Namely, to be accurate enough to test controller in realistic scenarios while remaining faster than real-time. This demanding tradeoff in terms of computational speed and simulation accuracy, led us towards symbolic simulation. For that reason, we interfaced YARP with Robotran.

Nevertheless, our objective is to let the user easily switch between each solutions. Therefore, a controller developed with the Robotran-YARP interface remains totally compatible with the Gazebo-Yarp simulator. Indeed, as the Gazebo-YARP plugin implements all the drivers that Robotran-YARP implements, the same controller can work with both. Also, YARP has the capability to interoperate with ROS. Therefore, one can also take advantage of the software developed by the ROS community.

Robotran is particularly tailored to fast and accurate multibody dynamics simulations [9]. Therefore, the proposed interface aims at testing controllers relying on dynamics (e.g. locomotion, whole-body control). Consequently, the drivers below were implemented in the Robotran environment.

### 3.1 Control Board Driver

On a real robot, control boards are electronic boards used to command the motors. They are made of joint control algorithms implemented at the firmware level. These low-level joints controllers should not be mistaken for the high-level user control algorithm. Typically, high-level algorithms do not directly drive the motors but rather send references to control boards that will, in turn, drive the motors. Therefore, control board driver receives reference commands from the user control algorithm and translates it into commands specific for the low-level joints controller. Different control modes are usually available (e.g. position or torque control). It also reads the joints state.

---

\(^2\)http://wiki.ros.org/gazebo_ros_pkgs
The main interfaces implemented for the Robotran Control Board driver are:

- **IEncoders**: gets the joints position, velocity and acceleration.
- **IOpenLoopControl**: sends a voltage command to the DC motors in open loop (without feedback control).
- **IPositionControl**: controls of the joint position. It can be a PID control (stiff mode) or a joint impedance control (compliant mode).
- **IVelocityControl**: controls of the joint velocity. It can be a PID control (stiff mode) or a joint impedance control (compliant mode).
- **ITorqueControl**: controls and measures the joints torques.
- **IControlMode2** and **IInteractionMode**: selects the control mode (position, velocity, torque, ...) and the interaction mode (stiff or compliant).

Others interfaces are used for changing the PID gains or for getting the motor electrical state variables.

On the Robotran simulator, different joints controllers are implemented mimicking the firmware low-level controllers of the real robot. The electrical model of a DC motor is also implemented through user derivatives [7],[8]. At each simulation time step, joints positions, velocities and torques are read by the driver. The desired joint positions or torques are then sent to the Robotran low-level controller.

### 3.2 Force-Torque Sensor Driver

Force-torque sensors measure the wrenches acting on a body.

This driver implements IAnalogSensor interface which outputs the 3 components of the force ($F_x, F_y, F_z$) and of the torque ($T_x, T_y, T_z$) expressed in a body fixed frame.

On a real robot this type of sensors are typically made of strain gauges measuring small deformation. In Robotran, this can be measured by adding 6 locked joints at the sensor location and measuring their wrench.

### 3.3 IMU Sensor Driver

The Inertial Measurement Unit (IMU) measures the orientation, velocity and acceleration of a body.

This driver implements IAnalogSensor interface which outputs the orientation (Euler angles), the angular velocity and the linear acceleration of a body. In Robotran this can be retrieve from a sensor added on the requested body of the robot model.

### 3.4 Clock Driver

This driver gets the simulation time in order to synchronize the controllers with the simulation; it is specific to simulator drivers. Indeed, on the contrary to real scenario, in simulation the time clock is not always constant. It might run faster or slower than real time. If the controllers and the simulator are not synchronized, they may have unrealistic behaviors. Therefore it is important that all the interconnected systems follow the same clock given by the simulator clock driver. It allows to test controller in a simulated real-time environment. Leveraging fast simulator like Robotran, tests of real-time controllers can even run faster than real time.
4 APPLICATIONS

Currently, the COMAN and WALK-MAN humanoids robots are interfaced with the Robotran-Yarp drivers. Furthermore, the drivers code is so generic that only text files of configuration need to be written to implement new mechatronic platforms.

This common interface allows to run the very same controllers on the robot and/or on the simulator. As an example, Figure 4 shows the Robot Motor Gui YARP module controlling the simulated COMAN. It is interesting to note that at the time this module was written, no interface between YARP and Robotran existed. Nevertheless, the module could be used without any modification. This illustrates the power of code reusing and collaboration brought by this framework. It offers a wide range of applications which can be presented in three groups, depending if the simulation is run before, during or after the tests on the real robot.

Classically, the simulator can be used before running the code on the real robot. This is the main reason for using a simulator. As stated in the introduction, testing the controller on a simulated environment before the real one offers to do fast code prototyping. It allows to test many situations and to optimize the controller. Also, a simulation environment provides information that might not be available on the real robot which is helpful to validate the code (e.g. a ground truth for a localization module). Furthermore, it is possible to write regression tests that directly interface with the simulator to verify that patches or new developments do not modify the expected behavior of robot software.

On top of that, the simulation can also run in parallel to the real robot (See Figure 5). Indeed, thanks to this modular interface, the same module can control a real robot and a simulated one (receiving copy of the commands). This provides an internal model particularly useful to perform state estimation and prediction. Among others, we plan to use this tool to support tele-operation with low bandwidth and noisy communication.

Finally, the simulation can be exploited after the real test to perform model identification. It can for example help to refine kinematic or dynamic parameters of the model.

![Figure 4](image.png)

Figure 4. The COMAN humanoid simulated in Robotran is controlled by a classical YARP-module, the "Robot Motor GUI" through the Control Board interface.
5 CONCLUSION AND FUTURE WORKS

Future works will consist in bringing the Robotran dynamics simulator and the interface plug-in into a generic library. It will further be extended to other platforms, like the iCub humanoid robots. We plan to use it to perform online system identification of the robots and its environment. Drivers for other sensors might also be implemented (tactile sensors, camera, etc).

In conclusion, a clean interface between the Robotran simulator and Yarp has been developed. It allows to improve and foster software developments among large collaborative project like the WALK-MAN project. We hope it can benefit to other researcher teams and therefore, we released the open source code in a public repository (https://gitlab.robotran.be/walkman).

Acknowledgments

This work was supported by the European Community’s 7th Framework Programme (FP7-ICT-2013-10) under Grant 611832 (WALKMAN collaborative project). We are grateful to Alexandra Zobova and Nicolas Van der Noot for their support in the development of the simulator.

REFERENCES


Verification of a Numerical Simulation Code for Underwater Catenary Chain

Xiang Q. Zhu*, Wan S. Yoo*

Department of Mechanical System Design, Pusan National University, 30 Jangjeon-dong Geumjeong-gu Busan, 609-735, South Korea.
zhuxiangqian@pusan.ac.kr

Department of Mechanical System Design, Pusan National University, 30 Jangjeon-dong Geumjeong-gu Busan, 609-735, South Korea.
wsyoo@pusan.ac.kr

ABSTRACT

Numerical simulation is an economical and effective method in the field of marine engineering. In this paper, a numerical simulation code for dynamic analysis of chain mooring is created using a new element frame which is based on vectors of element orientation and relative velocity. This paper aims at verifying the accuracy of this new simulation code through comparisons with real experiments and commercial simulation code. The real experiment is proceeded in a water tank using a small size chain which is controlled by an X-Z traversing device. Motions of the chain are recorded, and compared with simulation results by the numerical simulation code. What’s more, simulation results by the new simulation code for the large size chain are compared with those by commercial simulation code. The accuracy of the numerical simulation code for underwater chain mooring is verified by comparisons with both real experiments and commercial simulation codes.

Keywords: Numerical Simulation Code, Catenary Chain Modelling, Element frame, Dynamic Analysis, Verification.

1 INTRODUCTION

One common characteristic of marine facilities, no matter the floating platform or drilling system is the huge outline dimensions. Also, the complexity of the marine environment aggravates difficulties of real experiments. Therefore, virtual simulation codes are becoming popular in the field of marine engineering. The simulation code for chain mooring is almost based on the static analysis in former researches which ignore both the hydrodynamic forces and the added mass effects, and assumes the seabed being flat and horizontal [1-6]. Considering the range of motion, heavy weight in water and computational efficiency, it is reasonable to create the simplified chain modelling in past decades. With the development of composite materials and increasing computer processing speed, it is the time to conduct dynamic analysis of the underwater chain mooring. In other words, the hydrodynamic forces and added mass effects should be added to approximate reality. Lumped mass modelling is a favoured choice when doing dynamic analysis of towed cables [7-9]. Chain mooring is simplified using the lumped mass spring modelling scheme, wherein the chain is divided into N elements ordered from top to bottom, as shown in Figure. 1. One element frame is needed for each chain element to convert sustained forces to the inertial frame. The element frame in published researches is created based on Euler angle sets or the combination of Frenet frame Euler angle sets. However, singularities go with Euler angle sets when the first rotation is collinear with the third rotation, and calculation of the rotational transformation matrix is complex. Therefore, a new element frame is proposed for modelling catenary chain mooring in this paper. This new element frame is created based on vectors of element orientation and relative velocity of fluid [10, 11]. Formulations of both the rotational transformation matrix and hydrodynamic drag forces are expressed effectively with the new element frame. The numerical simulation code is established considering the stiffness and damping of the chain, apparent weight, hydrodynamic drag forces
and the effect of added mass. Accuracy is the key point for a virtual simulation code, so this numerical simulation code is checked through two ways: real experiments with small size chain mooring, and commercial simulation code with large size chain mooring. The real experiment is proceeded in a water tank, and one end of the chain is fixed with the tank and the other end is connected with an X-Z traversing device which moves with a constant speed. The water tank is made by transparent glass, so motions of the chain node can be recorded using a camera. The recorded video is proceeded by TEMA software which gives out the velocities and positions of marked points [12]. Finally, simulation results by the numerical simulation code are compared with those achieved from experiments. What’s more, the long chain mooring modelling is created in both the numerical simulation code and ProteusDS [13, 14]. X-direction wave and current are given during the dynamic simulation. The simulation results by the ProteusDS are compared with the numerical simulation code. Comparisons tell the numerical simulation code for chain mooring being created correctly.

Figure 1. Procedure for real experiment of chain modelling.

2 VERIFICATION WITH REAL EXPERIMENTS

2.1 Experiment setting

The real experiment is proceeded with a studless chain in a water tank (270 cm x 230 cm x 165 cm) with water depth of 143.5 cm. The upside of the chain is fixed on the guild-bar of an X-Z traversing device. The guild-bar can move X- and Z-directional with a given speed which is realised by continuous screw drives, as shown in Figure 2 (a). To make sure the chain does not touch the bottom of the tank, the other side of the chain is fixed above the bottom with a height of 72 cm. Six points are chosen and marked with respect to the chain span which is counted from the up end, shown in Figure 2 (b). Point 1 and Point 2 are marked on the guild-bar, the distance between these two points are 10 cm vertical. The moving speed of the X-Z traversing device is set as 0.5 cm/s and 1.0 cm/s X-directional, and the moving distance is 75 cm. The properties of the small size chain in real experiment are shown in shown in Table 1.
(a) X-Z traversing device on the water tank.               (b) Marked points on the chain mooring.

Figure 2. Experiment setting.

Table 1. Properties of chain in water tank.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Magnitude</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length</td>
<td>200</td>
<td>mm</td>
</tr>
<tr>
<td>Mass</td>
<td>1105</td>
<td>g</td>
</tr>
<tr>
<td>Elastic Modulus</td>
<td>211</td>
<td>Gpa</td>
</tr>
<tr>
<td>Drag Coef.</td>
<td>2.2</td>
<td></td>
</tr>
<tr>
<td>Added mass</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Diameter</td>
<td>9.5</td>
<td>mm</td>
</tr>
<tr>
<td>Element No.</td>
<td>20</td>
<td></td>
</tr>
</tbody>
</table>

2.2 Post-processing

Motions of the chain are recorded using a camera with 20 frames per second, the video tapes are processed using TEMA Automotive software which gives displacements of marked points, as shown in Figure 3. These values are given with respect to the coordinate in TEMA. These values are converted to obtain the real values in post-processing. The X-directional values are converted according to three values: firstly, the moving distance of Point A; secondly, the distance between Points A and O measured at the beginning; and the last is the distance measured at the end of experiment. While, the Z-directional values are conducted with three values: the distance of the Point 1 and Point 2 which are shown in Figure 2. (b); secondly, the distance between Points A and O measured at the beginning; and thirdly the distance between Points A and O measured at the end of experiment.
The experiment results are compared with those carried out by the numerical simulation code, as shown in Figure 5. There are some obvious differences between results by the numerical simulation code and those by the experiment on the Z-directional displacements of the Point G. What’s more, although the Point A traverses horizontally, the experimental results shows that the Point A drops slightly proportional with the traversing distance. Because the values of converted according to the distance between Points A and O, and distance between Points 1 and 2 which are far from the Point G, system error exist in the TEMA results, and proofreading is needed.

**2.3 Proofreading**

System error including the initial error and the method error, exists during recording the motions of the chain using a camera. Initial error is caused during locating positions of these marked points, which can be detected with measured values. The initial error of the Points G are shown clearly in Figure 4. The initial position in TEMA are proofread according the measured values in here. What’s more, because light refracts when passing through different mediums, and the caused error is proportional to the incidence angle simply. This method error can be proofread according to the Z-directional values of the moving Point A as shown in Figure 5. Correction values are added to all the Z-directional values of marked points based on the slope shown in Figure 5. Finally, the simulate results are compared with the real experiment and shown in Figure 6 and 7. There are still slight difference for Point B and C in Figure 6. These
differences remain the same magnitude from the beginning to the ending, which is caused by incorrect converting the recording values of marked Point B and C on the chain in TEMA.

Figure 5. The Z-directional value of Point A with a speed of 5 cm/s.

Figure 6. Positions of marked points with a speed of 5 cm/s.

Figure 7. Positions of marked points with a speed of 10 cm/s.
3 VERIFICATION WITH COMMERCIAL SIMULATION CODE

Commercial simulation code ProteusDS is one of widely used software for the dynamic analysis of mooring system and fishing net. Numerical modelling of chain in ProteusDS is established using a cubic-spline lumped mass method and shape function. The chain suffers a constant X-directional current and wave in this procedure. The wave height is 2.4 m and wave period is 8 s, the current is a constant speed with a value of 1 m/s. An upward load acting on the first node as loads from the floating platform. Table 2 shows the properties of the large chain mooring in ocean, and the simulation is proceeded 300 seconds.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Magnitude</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st Node Position</td>
<td>[50;0,-10]</td>
<td>m</td>
</tr>
<tr>
<td>21st Node Position</td>
<td>[0;0,-500]</td>
<td>m</td>
</tr>
<tr>
<td>Density</td>
<td>7800</td>
<td>kg/m³</td>
</tr>
<tr>
<td>Length</td>
<td>500</td>
<td>m</td>
</tr>
<tr>
<td>Elastic Modulus</td>
<td>211</td>
<td>Gpa</td>
</tr>
<tr>
<td>Drag Coef.</td>
<td>1.2</td>
<td></td>
</tr>
<tr>
<td>Added mass</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Diameter</td>
<td>0.03</td>
<td>m</td>
</tr>
<tr>
<td>Element No.</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>Upward Loads</td>
<td>3.5e4</td>
<td>N</td>
</tr>
</tbody>
</table>

The simulation results of the 1st, 6th, 10th and 15th node are shown here. Displacements of the chosen nodes by the ProteusDS are compared with those by the numerical simulation code in Figure 8. It is hard to notice the slight differences due to the large size of chain. These differences, however, could be detected by comparing the Z-directional displacements of the 1st node as shown in Figure 9. Chain by the numerical simulation code is lower than that by the commercial simulation code by 1 meter. Considering the total length of the chain is 500 meters and the water depth is 500 meters, these differences are acceptable. Figure 10 illustrates the X-displacements of the chosen nodes. Wave loads have more effect on motions of the X-direction than those of the Z-direction, and this effect decreases with increasing of the water depth.

![Figure 8. Displacements of chosen nodes during the simulation.](image)
4 CONCLUSIONS
The numerical simulation code for chain mooring is verified through comparisons with both the
real experiments and commercial simulation code. The real experiment is proceeded in a water
tank with a small size catenary chain. The simulation results match well with the experimental
results. A large size chain modelling is created in both commercial simulation code and the
numerical simulation code. The simulation proceeds 300 seconds with X-directional linear wave
and current. This numerical simulation code also shows high accuracy on large size chain
mooring through comparison with commercial simulation code. This paper also proofreads the
system error of the experiment data using measured initial values. The method of processing the
experiment data can be used for reference.

ACKNOWLEDGEMENTS
This work was supported by the Human Resources Development program (No. 20134030200290) of the Korea Institute of Energy Technology Evaluation and Planning (KETEP) grant funded by the Korea government Ministry of Trade, Industry and energy.
REFERENCES


Analytical and Numerical Multibody Dynamic Modeling and Constraint Force Analysis in an Axial Piston Pump

T. R. Milind*,M. Mitra#

* Senior Engineer
Eaton India Engineering Center
Kharadi, Pune, India
TRMilind@eaton.com

# Associate Professor
Department of Aerospace Engineering
Indian Institute Technology Bombay
Powai, Mumbai, India
mira@aero.iitb.ac.in

Abstract

Multibody dynamic modeling is an important area in computational dynamics which is useful in modeling complex systems consisting of many components interconnected by joints and forces. The modeling of rotating machines is a major area of study in multibody dynamics. Multibody methods are especially useful in modeling complex rotating systems wherein rotor consists of parts which exhibit a large translation/rotation relative to it. The swash plate type axial piston pump is an important type of rotating machine which is widely used in hydraulic systems. In the present work, the modeling of an axial piston pump is carried out using a multibody dynamics framework considering the 3D dynamics of the major components in the pumping mechanism and the pressure forces acting due to the hydraulic fluid. The equations of dynamic equilibrium for the different components are developed using the augmented Newton-Euler formulation. The pumping mechanism dynamics is formulated as an inverse dynamic problem and closed-form solutions for the constraint reactions are obtained. The analytical models are developed for the general case of the axial piston pump mechanism having N pistons. A numerical model is developed using the equations of motion in the augmented Newton-Euler formulation with Lagrange multipliers. The closed form solutions are compared with numerical simulations. The time variation and the frequency spectrum of the constraint reactions are studied for the case of a nine pumping piston configuration.

Introduction

Multibody dynamic modeling is an important area in computational dynamics which is useful in modeling complex systems consisting of many components interconnected by joints and forces. The modeling of rotating machines is a major area of study in multibody dynamics. Multibody methods are especially useful in modeling complex rotating systems wherein the rotor consists of parts which exhibit a large translation/rotation relative to it. The determination of constraint forces and its time variation is needed in the design of rotating systems and improvement in the durability, vibration and noise performance. The forward and inverse dynamics are two important classes of analysis in the study of dynamical systems. In most applications the forward dynamic analysis which involves solution of equations of motion and numerical time integration is used to determine the motion and forces.

The swashplate type axial piston pump is a type of rotating machine which is widely used in hydraulic power systems. There are several studies carried out on the modeling and analysis of axial piston pumps. The 1D flow models are developed to study the pressure ripples and leakage [1,2]. The 3D modeling of the fluid flow using Navier-Stokes or Reynolds equations and CFD or finite difference solution techniques is used to study the flow behavior at a specific interface [3,4]. The coupled 3D flow and mechanical dynamic models at a specific interface are helpful in lubrication studies [5,6,7]. The dynamic modeling of the pump using 1D models or Bond-Graph techniques enables the study of the swashplate motion and the controls performance [8,9,10,11]. In addition to the above, the structural design and optimization, estimation of fatigue life, and improvement in noise and vibration are important considerations in the pump design. The 3D dynamic modeling and analysis of the constraint reaction forces [12,13] is critical for addressing these issues. The kinematic modeling [14,15] of the pumping mechanism is an important step in the overall dynamic modeling of the pump. The multibody dynamic method provides an ideal framework for modeling the dynamics of the pump [16,17].

In the present work, the methodology of modeling an swashplate type axial piston pump in a multibody dynamics framework is developed. The closed form solutions for the time variation of
constraint reactions and study of their frequency spectra are not previously developed in the literature to the best knowledge of the authors. The analytical model will be helpful in the design of the control system and inverse problems as opposed to the numerical scheme. The modeling is carried out using a multibody dynamics framework considering the 3D dynamics of the major components in the pumping mechanism and the pressure forces acting due to the hydraulic fluid. The equations of dynamic equilibrium for the different components are developed using the augmented Newton-Euler formulation [18]. The pumping mechanism dynamics is formulated as an inverse dynamic problem and closed-form solutions for the constraint reactions are obtained. The analytical models are developed for the general case of the axial piston pump mechanism having N pistons. The numerical model is developed using the equations of motion in the augmented Newton-Euler formulation with Lagrange multipliers[19]. The closed form solutions are compared with numerical simulations. The time variation and the frequency spectrum of the constraint reactions are studied for the case of a nine pumping piston configuration.

Pump Description

Figure 1 shows the general configuration of an axial piston swashplate type hydrostatic pump. The pump consists of pistons within a common cylindrical block. The pistons are positioned in circular array within the cylinder barrel at equal intervals about the input shaft (x-axis). The cylinder barrel has a splined connection with the driveshaft.

The cylinder barrel is held tightly against a valve plate using the force of the compressed cylinder barrel spring and pressure in the cylinder barrel bore. A thin film of oil separates the valve plate from the cylinder block which forms a hydrostatic bearing between the two parts. A ball-and-socket joint connects the base of each piston to a shoe. A hydrostatic bearing surface separates the shoe from the swashplate. While the valve plate is held in a fixed position, the cylinder barrel is driven about the shaft axis at a constant angular speed. During this motion, each piston periodically passes over the inlet and outlet ports on the valve plate. Since, the shoes are held against the inclined plane of the swashplate, the pistons undergo an oscillatory displacement parallel to the axis of the driveshaft.

Analytical modeling of the pump dynamics

The analytical modeling of the pump dynamics is carried out in three principal steps viz., kinematic modeling, dynamic modeling, and development of the closed form solutions. The dynamic modeling is carried out by developing the differential EOMs using the augmented Newton-Euler formulation where the constraint reaction forces are explicitly considered. The equations are developed for the individual components in the form,

$$\left[ M^i \right] \{ \ddot{q}^i \} - \{ Q_e^i \} = \{ F_e^i \} \quad (1)$$
Where $[M']$ is the mass-inertia tensor, $\{q'\}$ is the vector of generalized coordinates, $\{Q_c\}$ is the constraint force vector and $\{F_e\}$ is the external force vector.

The piston in the axial piston pump is subjected to the external pressure force ($F_{ex}^i$) which acts along the X direction. The normalized pressure force considered is shown in Figure 2.

![Figure 2](image)

**Figure 2** A representative pressure force variation

The constraint reaction forces and moments on the $i^{th}$ piston are exerted by the $i^{th}$ shoe and the cylinder barrel. Thus, the equation describing the dynamic equilibrium of the $i^{th}$ piston is given by,

$$
\begin{align*}
\begin{bmatrix}
M_p & 0 & 0 \\
0 & M_p & 0 \\
0 & 0 & I_p
\end{bmatrix}
\begin{bmatrix}
q_p' \\
\theta_p'
\end{bmatrix}
&=
\begin{bmatrix}
R_{px} \\
R_{py} + R_{pz} \\
M_{px} + M_{py} + M_{pz} - R_{pxl} - R_{pyl}
\end{bmatrix}
-
\begin{bmatrix}
-F_{ex}^i \\
0 \\
0
\end{bmatrix}

\tag{2}
\end{align*}
$$

Each term in the Equation (2) and following equations is explained in the nomenclature list. Similarly, the equations of dynamic equilibrium for the shoes, cylinder barrel and swash plate are developed.

The kinematic modeling is carried out to determine the positions of the piston center of mass, shoe center of mass, center of the ball-and-socket joint between piston and shoe, and the center of the shoe-swash plate interface in the global reference frame XYZ (Figure 1). The modeling is carried out by treating the swash plate angle as constant, considering steady operating speed and not considering the clearances/interaction with lubricating fluid films. The variation of the positions of the center of mass of the $i^{th}$ piston in the global reference frame resulting from the position analysis is given by,

$$
\{r_p\} =
\begin{bmatrix}
l_{min} + \frac{D}{2} \tan(\alpha) \sin(\omega t + \pi + (i - 1) \frac{2\pi}{N}) \\
\frac{D}{2} \cos(\omega t + \pi + (i - 1) \frac{2\pi}{N}) \\
\frac{D}{2} \sin(\omega t + \pi + (i - 1) \frac{2\pi}{N})
\end{bmatrix}
\tag{3}
$$

The expressions for the center of the ball-and-socket joint between piston and shoe, shoe center of mass, and shoe-swash plate interface are obtained by adding the vectors $\begin{bmatrix}
-l_{min} & 0 & 0 \\
-l_{min} + (l_{shc2} - l_{shc1}) \cos \alpha & 0 & (l_{shc2} - l_{shc1}) \sin \alpha
\end{bmatrix}^T$ and $\begin{bmatrix}
-l_{shc1} \cos (\alpha) & 0 & l_{shc2} \sin (\alpha)
\end{bmatrix}^T$ to $\{r_p\}$. The expressions for velocity and acceleration are obtained by differentiating the Equation (6) with respect to time. The expressions for the position of the shoe-swash plate interface in a rotated coordinate system $X'Y'Z'$ (rotated by $\alpha$) shows that the interface moves along an ellipse on the swash plate surface with the semi-minor axis $(D/2)$ and semi-major axis $(D/(2 \cos \alpha))$.
The closed form expressions for the different constraint reactions viz., horizontal swashplate bearing force ($R_{sbX}$), the vertical swashplate bearing force ($R_{sbZ}$), control piston force ($R_c$), horizontal swashplate bearing moment ($M_{sbX}$), vertical swashplate bearing moment ($M_{sbZ}$), reaction forces at driveshaft-cylinder barrel interface ($R_{bsY}$, $R_{bsZ}$), and moments at the driveshaft-cylinder barrel interface ($M_{bsY}$, $M_{bsZ}$) are obtained by solving the dynamic equilibrium equations. The forces and moments on the swash bearings are obtained at the center of the swashplate which is coincident with the origin of the global coordinate system shown in Figure 1. The Equations (4)-(12) give the expressions for the forces and moments exerted by the swashplate on the bearings and the control piston.

\[ R_{sbX} = \sum_{i=1}^{N} \left( -m_i \omega^2 D \left( m_i \alpha + F_{pi} \right) \right) \tag{4} \]

\[ R_{sbZ} = \sum_{i=1}^{N} \left\{ \left( F_{pi} - F_{pi} \right) \tan \alpha + \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \right\} \tag{5} \]

\[ R_c = \sum_{i=1}^{N} \left\{ \left( F_{pi} - F_{pi} \right) \tan \alpha + \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \right\} \tag{6} \]

\[ M_{sbX} = \sum_{i=1}^{N} \left\{ \left( l_{pm1} \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \right) \cos \alpha \right\} \tag{7} \]

\[ M_{sbZ} = \sum_{i=1}^{N} \left\{ \left( l_{pm1} \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \frac{1}{l_{pm1}} \right) \cos \alpha \right\} \tag{8} \]

And, the instantaneous angular position of the $i^{th}$ piston is given by:

\[ \theta_i = \omega t + \pi + \frac{2\pi}{N} \]

The cylinder barrel has a splined connection with the driveshaft and the constraint reactions on the driveshaft are obtained on the point at the center of the splined connection. The Equations (9)-(12) provide the expressions for the forces acting on the driveshaft.

\[ R_{bsY} = \sum_{i=1}^{N} \left( F_{pi} - F_{pi} \right) \cos \alpha \] \tag{9}

\[ R_{bsZ} = \sum_{i=1}^{N} \left\{ \left( F_{pi} + F_{pi} \right) \tan \alpha - F_{pi} \right\} \] \tag{10}

\[ M_{bsY} = \sum_{i=1}^{N} \left\{ \left( F_{pi} + F_{pi} \right) l_{pm1} \tan \alpha - \left( \left( F_{pi} + F_{pi} \right) \tan \alpha - F_{pi} \right) \left( x_p - l_{brl1} \right) \right\} \] \tag{11}

\[ M_{bsZ} = \sum_{i=1}^{N} \left\{ F_{pi} l_{pm1} + \left( F_{pi} - F_{pi} \right) \left( x_p - l_{brl1} \right) \right\} \] \tag{12}
The Equations (4)-(12) represent the basic set of constraint forces present in the pumping mechanism. However, for practical applications the knowledge of forces on the individual driveshaft and swashplate bearings is required explicitly. These forces are the vertical end cover bearing force \( R_{ebZ} \), vertical housing bearing force \( R_{hbZ} \), the horizontal high pressure swashplate bearing force \( R_{hpX} \), the horizontal low pressure swashplate bearing force \( R_{lpX} \), the vertical high pressure swashplate bearing force \( R_{hpZ} \), the vertical low pressure swashplate bearing force \( R_{lpZ} \), and control piston force \( R_c \). These can be obtained from the above basic set of constraint reactions.

**Numerical Modeling**

As mentioned earlier a numerical model is developed to validate the analytical closed form expressions. In the numerical modeling of the pump dynamics, the governing dynamic Equations are developed using augmented Newton-Euler formulation with Lagrange multipliers. This formulation leads to symmetric coefficient matrices with a sparse structure. The general form the equations is given by,

\[
\begin{bmatrix}
M & C_q^T \\
C_q & 0
\end{bmatrix}
\begin{bmatrix}
\ddot{q} \\
\lambda
\end{bmatrix} = 
\begin{bmatrix}
F_e \\
Q_d
\end{bmatrix}
\tag{13}
\]

Where \( [M] \) is the mass-inertia tensor, \( [C_q] \) is the constraint Jacobian matrix, \( \{ \ddot{q} \} \) is the acceleration vector, \( \{ \lambda \} \) is the vector of Lagrange multipliers, \( \{ F_e \} \) is the vector of external forces, and \( \{ Q_d \} \) is the quadratic velocity vector. The numerical model is developed for a nine pumping piston configuration. The equations of motion are solved, using numerical integration based on the implicit Newmark scheme, to determine the position \( \{ q \} \) and Lagrange multipliers \( \{ \lambda \} \). Further, the Lagrange multipliers \( \{ \lambda \} \) are used to determine the constraint reaction forces \( \{ Q_c \} = -[C_q]^{-1}\{ \lambda \} \). The numerical simulations are performed at constant RPM over one complete revolution of the drive shaft.

**Results and Discussion**

The closed form solutions are compared with the numerical results for the single piston and nine piston configurations. Figure 3 shows the comparison of horizontal swashplate force obtained using analytical and numerical models for the single piston configuration. It is observed that there is a close agreement between the results of analytical and numerical models. Similar match has been observed for all other constraint forces. This validates the analytical closed form expressions.

![Figure 3 Comparison of the numerical and analytical model](image)
Figure 4 shows the variation in the forces for the nine piston configuration. Figure 4(a)-(c) show the variation in the bearing forces and Figure 4(d) shows the variation of the control piston force.

Figure 6 Bearing and Control piston forces in a nine piston configuration

The individual bearing reactions are important from the practical considerations and are computed from the constraint reaction forces and moments. Figure 5 shows the frequency domain results of the different bearing reactions for the nine piston model. The bearing reactions are dominant in the pumping frequency and its multiples.

Figure 5 Frequency domain spectrum of bearing and control piston forces the nine piston configuration

In the nine piston model since the pumping frequency is the 9th order, the bearing reaction and control piston forces are dominant in the 9th order and its multiples. Thus, for the present case of the pressure force variation, it is seen that the driveshaft bearing forces and the horizontal swashplate forces are having a dominant 9th order. The vertical high pressure (HP) swash bearing force is
dominant in the 18th order while the vertical low pressure (LP) swash bearing force is dominant in the 9th order. The control piston force is dominant in the 18th order.

In the present work a novel analytical approach of modeling the dynamics and predicting the constraint forces in an axial piston pump is developed. The approach considers the all the major internal components in the pump, 3D dynamics of the pumping mechanism and fluid pressure forces acting on the piston. The constraint reaction forces are helpful in structural design and optimization, identification of worst case loading scenarios, fatigue life estimation and improvement in noise and vibration performance. The pressure ripple and torque ripple while provide a qualitative indication of the overall noise, these are still single, scalar, and indirect measures of the noise generation. The set of constraint reactions provide a better representation for optimizing noise performance as in addition to providing magnitudes of the loading they provide their directions, locations and phases, as well.

The closed form solutions are developed for the general case of N pumping pistons. The expressions provide complete time variation of the forces and explicitly show the dependence on the design parameters. The closed form solutions were found to provide the influence of swashplate angle/flow output, valve plate designs and rotating group designs on the static loading and dynamic excitation forces in the pump.

The analytical closed form solutions closely agreed with the numerical simulations. The analytical solutions were used to study the practical case of nine piston pump considering a representative pressure force variation. The constraint reaction forces were analyzed in time domain and frequency domain. The overall approach is found to be helpful in identifying the dominant constraint forces as well as the dominant orders.

The multi-body dynamic modeling is currently being further developed by integration of the pump control dynamics and inclusion of the component flexibility.

References


**NOMENCLATURE**

\[
\begin{bmatrix}
    r_p^i \\
    \theta_p^i
\end{bmatrix}
\]

Position vector of the \(i^{th}\) piston center of mass

\[
\begin{bmatrix}
    r_{sh}^i \\
    \theta_{sh}^i
\end{bmatrix}
\]

Position vector of the \(i^{th}\) shoe center of mass

\(I_{pml}\)

Distance between the piston center of mass and center of ball-and-socket joint

\(I_{shel}, I_{she2}\)

Dimensions of the Shoe

\(I_{swal}, I_{swa2}\)

Dimensions of the Swashplate

\(I_{brl}\)

Distance from the global origin to the center of the spline of the driveshaft

\(I_{brl2}\)

Distance from the center of the driveshaft spline to the center of cylinderbarrel bore

\(I_{shafl}\)

Distance from center of driveshaft spline to center of housing bearing

\(I_{shafl2}\)

Distance from center of driveshaft spline to center of end cover bearing

\(F_{px}^i, F_{py}^i, F_{pz}^i, F_{x}^i, F_{y}^i, F_{z}^i\)

Inertial forces of \(i^{th}\) piston and shoe

\(R_{pxX}^i, R_{pxY}^i, R_{pxZ}^i\)

Reactions at \(i^{th}\) piston-shoe interface
Reactions and moments at $i^{th}$ piston-barrel interface

$R_{pbX}^i, R_{pbZ}^i, M_{pbX}^i, M_{pbZ}^i$

Reactions and Moments at $i^{th}$ shoe-swashplate interface

$R_{ssX}^i, R_{ssZ}^i, M_{ssX}^i, M_{ssZ}^i$

Reaction force and moments on Swash Bearing along/about X and Z direction

$R_{shX}^i, R_{shZ}^i, M_{shX}^i, M_{shZ}^i$

Control Piston Force

$R_{bsX}^i, R_{bsZ}^i, M_{bsX}^i, M_{bsZ}^i$

Reaction force and moments at driveshaft barrel spline along Y and Z direction

$F_{pr}^i(t)$

Reaction force at driveshaft barrel spline about Y direction

$m_{p}^i, m_{sh}^i, m_{shb}^i, m_{sw}^i$

Moment on drive shaft at barrel shaft spline about Y direction

$\alpha$

Swash plate angle

$L$

Stroke of the piston

$\omega$

Angular frequency of rotation of the shaft

$T$

Time

$D$

Pitch circle diameter of the bores in the cylinder barrel

$XYZ$

Global coordinate system

$X'Y'Z'$

Coordinate system rotated by swashplate angle about Y axis with respect to global coordinate system
Development of 3D multibody models of an Intervention-Autonomous Underwater Vehicle (I-AUV)

B. Allotta*, R. Conti*, R. Costanzi*, F. Fanelli*, E. Meli*, A. Ridolfi*

* Department of Industrial Engineering (DIEF)
University of Florence
Via di S. Marta 3, 50139 Florence, Italy
roberto.conti@unifi.it

ABSTRACT

Intervention Autonomous Underwater Vehicles (I-AUVs) represents one of the most challenging frontier of robotics. To develop new control architectures, a precise modelling of the mechanisms involved in the manipulation tasks is needed. The focus of this paper is the multibody modelling and the control of an Intervention-Autonomous Underwater Vehicle (I-AUV). An accurate model of the whole system has been developed, including vehicle-fluid interaction. A suitable 3D contact model has been developed for the contact between the gripper and the object to be manipulated. A control strategy for the whole I-AUV system is proposed, comprising a suitable grasp planning strategy. Finally, an evaluation of the I-AUV control system performances have been carried out.

Keywords: Multibody models, Underwater Manipulation, Autonomous Underwater Vehicles

1 Introduction

Nowadays, a considerable number of operations in sea-rescue, research and maintenance of oil rig appliances have been carried out by robotic devices applied on underwater vehicles. In particular, a considerable number of tasks need manipulation capabilities to achieve success [1], [2]. Currently, the standard in the underwater field is constituted by Remotely Operated Vehicles (ROVs) equipped with one or more robotic arms (Intervention-ROVs) [3], employed especially at high depths. These vehicles can be tele-operated for long periods of time and are usually controlled by means of a master-slave approach [4], [5]. However, the use of I-ROVs has several drawbacks and limitations: for instance, the costs of the missions are usually very high; in addition, delays in the control loop may happen (due to the physical characteristics of the communication channel) and the ROV operator must receive a special training. To overcome ROVs limitations, Autonomous Underwater Vehicles (AUVs) have been developed; these vehicle are able to execute missions in the underwater environment without human support. Intervention-AUVs (I-AUVs), i.e. AUVs equipped with one or more robotic arms, have been introduced to autonomously complete underwater manipulation tasks [3]. Nonetheless, I-AUVs are still characterized by many open problems, such as the control of the vehicle-manipulator system (especially when relevant vehicle velocities are considered) or the cooperation among vehicles. Particularly, autonomous free floating base underwater manipulation still represents a theoretically and technologically challenging topic, principally in the presence of relevant vehicle velocities (i.e. dynamic manipulation, in contrast with hovering manipulation).

In this paper, a detailed 3D multibody model of the I-AUV system (vehicle, arm, gripper, object to be manipulated, contact model and fluid interaction [6], [7]) has been developed to test the proposed control strategy. A suitable 3D contact model has been developed for the contact between the gripper and the object to be manipulated. For what concerns the control technique, a decoupled vehicle-manipulator strategy has been employed [1], [3]. This kind of techniques offers simpler hardware implementation and is more robust against...
the knowledge of the system parameters with respect to arm-vehicle coupled strategies. In addition, exploiting the hand kinematics, the control of the gripper has been further decoupled from the arm control: this way, the performances of the I-AUV are improved while maintaining higher vehicle velocities. Furthermore, a grasp planning algorithm, based on optical cameras [8], is proposed.

To validate the effectiveness of the proposed approach, a suitable test case has been simulated using the software Matlab®. As regards future developments, the proposed technique will be used in appropriate hardware tests in the framework of existing projects coordinated by the Mechatronics and Dynamic Modelling (MDM) Lab of the Department of Industrial Engineering of the University of Florence, such as SUONO project.

2 I-AUV multibody modelling

2.1 The I-AUV system

The AUV possesses 6 Degrees of Freedom (DOFs) and is equipped with a serial 7 DOFs manipulator arm with a spherical wrist. On top of the wrist a 6-DOFs gripper is mounted, constituted of 3 fingers, each one composed of 2 phalanxes connected by rotational joints. Figure 1 shows the reference frames linked to each rigid body and used to compute the hydrodynamic terms. The study of the vehicle model has been separated from the analysis of the manipulation system; furthermore, it has been assumed that the gripper is rigidly connected to the robotic arm.

2.2 Vehicle model

The considered AUV model is the one proposed in [6]. The dynamics of the vehicle is expressed in terms of vectors $\eta \in \mathbb{R}^6$ (which contains the position and the orientation of the AUV with respect to a fixed NED frame) and $\nu \in \mathbb{R}^6$ (composed of the linear and the angular velocity of the vehicle expressed in a frame attached to the center of gravity of the AUV), and it is governed by the following vector equations [6]

$$M\ddot{\nu} + C(\nu)\nu = \tau_H(\nu, \nu_C) + g_\eta(\eta) + \tau.$$  

(1)

$M$ and $C(\nu)$ are the mass matrix and the Coriolis and centrifugal effect matrix, $g(\eta)$ and $\tau$ are the contribution due to the gravity effects and to the external forces and moments applied to the vehicle as to the body frame. These contributes are referred to the rigid body characteristics. In order to use the classic multibody modelling techniques, the hydrodynamic effects $\tau_H(\nu, \nu_C)$ are partially decoupled from the dynamical equations. Buoyancy and hydrodynamic effects are introduced into the model by means of generalized Lagrangian forces applied to each body of the system. Splitting $\nu$ into the sum of the
current velocity $v_C$ and of the relative velocity $v_r$ ($v = v_C + v_r$) [6], the following expression for $\tau_H(v_C)$ can be extracted [9]:

$$\tau_H = -M_A \dot{v}_r + C(v_r) v_r + C(v_c) v_c + C_A(v_r) v_r - D(v_r) v_r.$$  \hspace{1cm} (2)

$M_A$ is the added mass matrix due to the fluid viscosity, $C_A$ is the Coriolis and centrifugal added effects matrix, and $D(v_r)$ is the damping matrix.

### 2.3 Manipulation system model

The manipulation system is composed of the robotic arm and of the gripper. The former is a serial manipulator with 7 DOFs (shoulder roll and pitch, elbow roll and pitch, RPR spherical wrist). The dynamic model of the arm is simulated through multibody techniques, in which each rigid body is modelled as follows:

$$M^I \ddot{q}^I + C^I(q, \dot{q}) + g(q) + J^T_f \dot{h}_c = \tau_f .$$  \hspace{1cm} (3)

The meaning of the terms is the same of Eq. (1); the superscript $^I$ indicates each link of the arm, while the subscript $f$ is introduced to distinguish between AUV and link quantities. For what concerns the gripper, each finger is locally equivalent to a planar 2-DOFs manipulator; denoting with $q_i$ the vector of joint coordinates of the $i$-th finger, the dynamics of each finger is expressed by [1]:

$$B(q) \ddot{q} + C(q, q) + g(q) + J_f^T h_c = \tau_f ,$$  \hspace{1cm} (4)

where $B$ is the inertia matrix, $C(q, \dot{q})$ and $g(q)$ include centrifugal, Coriolis and gravitational effects, $J_f$ is the finger Jacobian matrix, $h_c \in \mathbb{R}^{6 \times 1}$ is the vector of forces/torques due to interactions with the environment and $\tau_f$ are the joint torques. As for the vehicle and the arm, multibody modelling techniques have been used for the gripper.

### 2.4 3D Contact model

A complete 3D contact model has been derived. Hard finger contact has been assumed [1], [2], [10], [11]. The algorithm used to compute contact forces can be divided into two steps: 1) Contact Point Detection, i.e. the computation of the position of the contact points and their distance and 2) Contact Force Computation, i.e. the computation of the force exerted at the contact surface. Figure 2 shows the notation adopted to describe a generic contact between the gripper fingertips and an object to be manipulated. $c_o, R_o$ denote the pose of the object in the fixed frame; $p_f, p_o$ and $v_f, v_o$ are the position and the velocities of the contact points on the fingertip and on the object, and $D = p_f - p_o$ and $s = v_o - v_f$ denote the distance between the contact points and the sliding between the surfaces in contact. $N$ is the contact normal. If contact occurs (i.e. the penetration $\rho = D^T N$ becomes negative), normal and tangential contact forces are computed as follows:

- Normal force follows the spring-damper model ($k_n > 0$, $c_n > 0$):
  $$f_N = (k_n \rho - c_n s_N) N, \quad s_N = s^T N ;$$  \hspace{1cm} (5)

- Tangential force is given by:
  $$f_t = -\mu ||f_N|| \frac{s_t}{||s_t||}, \quad s_t = s - s_N N, \quad \mu(||s_t||) = \mu_k + (\mu_s - \mu_k) e^{-k||s_t||},$$  \hspace{1cm} (6)

for static and kinematic friction coefficients $\mu_s > \mu_k$ and a tunable parameter $k > 0$. 

906
3 I-AUV control strategy

A vehicle-manipulator control strategy has been adopted, in view of its simpler hardware implementation and greater robustness with respect to the unknown parameters of the system compared to centralized I-AUV control techniques [3]. In addition, exploiting the hand kinematics, the gripper control has been further decoupled from the arm control. The whole control system follows a “backward strategy”: grasp planning algorithms compute the position of optimal contact point on the object, and a suitable reference trajectory for the fingers is generated; then, the reference trajectory for the arm is computed (starting from the desired pose for the fingers). The applied control is in charge of following these trajectories closely. Finally, an admissible reference for the AUV is computed. It has been assumed that all the DOFs of the system are controlled.

3.1 Trajectory generation of the manipulation system

The desired values of the reference trajectory of the fingers are the contact points on the object; once the reference for the fingers has been computed, a suitable reference trajectory for the arm (which allows the fingers to reach the desired position) is then generated. For what concerns the trajectory itself, it is computed as linear combination of initial and desired values. The procedure is different for position and orientation reference:

- Position trajectory generation: let \( x \) denote the generic (scalar) position variable; its desired trajectory is chosen as:

\[
x(t) = \lambda(t)x_i + (1 - \lambda(t))x_d,
\]

where \( x_i \) and \( x_d \) represent the initial and the desired value of \( x \) and \( \lambda(t) \) is a parameter that continuously varies from 1 to 0. The smoothness of the trajectory is maintained even if \( x_d \) varies with time;

- Orientation trajectory generation: initial and desired orientations are expressed as unit quaternions \( q_i \) and \( q_d \); then, Spherical Linear intERPolation (SLERP) is applied, in order to compute a constant angular velocity rotation:
Figure 3. Dynamic manipulation task performed

\[ q(t) = \frac{\sin(1 - \lambda(t))}{\sin(\theta)} \mathbf{q}_i + \frac{\sin(t\theta)}{\sin(\theta)} \mathbf{q}_d, \quad \theta = \cos^{-1}\left(\mathbf{q}_r^T \mathbf{q}_d\right). \quad (8) \]

Regarding control, each finger is independently controlled with an inverse dynamics position control [12], while the arm is kinematically controlled, in order to overcome the problem of obtaining exact values for its hydrodynamic coefficients. A Closed Loop Inverse Kinematics Control (CLIK) [1] has been used, exploiting the arm’s redundancy to maintain the manipulator far from singularities. The desired joint values obtained from the CLIK algorithm constitute the reference for seven PID controllers, each one applied to a single DOF of the arm.

3.2 Trajectory generation of the AUV

The reference trajectory of the AUV is generated so as to constantly maintain the object to be manipulated inside the arm and gripper workspace. Regarding control, a decoupled PID control strategy has been adopted, using a SISO PID controller for each DOF of the vehicle [6].

4 Numerical simulations

A dynamic manipulation task has been simulated to validate the behaviour of the I-AUV system; the task consists in grasping a cylinder lying on the seabed. The vehicle starts from rest and is accelerated until it reaches steady-state speed; when the cylinder enters the field of view (FOV) of an eye-in-hand camera, the reference trajectory is changed in order to align the camera focal axis with the line connecting the palm of the gripper to the estimate of the position of the geometrical centroid of the object. After the grasp, the cylinder is lifted and the arm reaches a final rest configuration. The AUV never decelerates during manipulation. The scheme of the planned task is reported in Figure 3. The described mission has been simulated in a Matlab®-Simulink® environment. Three simulations have been carried out, at different vehicle speeds: 0.1 m/s, 0.2 m/s and 0.25 m/s, in order to analyse the effect of increasing speed on the performances of the control system. Figure 4 reports the 3D trajectory of the AUV, of the gripper, of the fingertips and of the cylinder obtained during the fastest simulation. For what concerns the arm, the position error increases as the AUV moves faster; however, even in the worst case it is kept small during the execution of the manipulation task. Figure 5 shows the position of the center of gravity of the cylinder on each axis. It is easily recognizable the time when the object is grasped and then lifted (motion on the \(x\)-axis and on the \(z\)-axis, respectively), while transversal motion is negligible.

5 Conclusions and future developments

A detailed multibody model of an I-AUV is proposed in this paper. In particular, to better analyse the effectiveness of the multibody models, the most challenging autonomous manipulation (dynamic manipulation) has been considered. Dynamic manipulation denotes
manipulation tasks executed while the vehicle maintains relevant velocities, further complicating the execution of the mission due to the dynamic interaction between the AUV and the manipulator. A complete multibody model of the I-AUV system has been derived, including interaction with the fluid and contact with the object to be manipulated. The I-AUV is controlled by means of a decoupled vehicle-manipulator strategy, further decoupling the control of the gripper exploiting the hand kinematics. Different relative speeds between the I-AUV and the object, in the same simulation scenario, have been simulated with satisfying results, showing how the developed multibody models and the adopted strategy allow the execution of the task. As concerns future in-
vestigations, different simulation scenarios are required to establish the maximum velocity that can be maintained during the manipulation phase. Further improvements are scheduled, with special attention given to data acquisition and to autonomous calculations, before the application of the proposed strategy in the framework of the Italian project SUONO and of the FP7 European project ARROWS.

6 Acknowledgement

This work has been partially supported by the European ARROWS project, that has received funding from the European Union’s Seventh Framework Programme for research, technological development and demonstration under grant agreement no 308724.

REFERENCES


Dynamics of the Multimass Rotor on Active Hybrid Bearings

Roman N. Polyakov*, Sergey V. Mayorov#, Maxim E. Bondarenko*, Leonid A. Savin*

*Department “Mechatronics and International Engineering”, State University ESPC, Orel, Naugorskoe shosse, 29, 302020, Russia
romanpolak@mail.ru

#Research and Software Development Centre APM, Moscow region, Korolev, Oktyabr’sky boulevard 14, office 6, 141077, Russia,
sergeyostu@yandex.ru

ABSTRACT

The paper focuses on the results of studying the dynamic behavior of a multi-mass asymmetric rigid rotor on the active hybrid bearings (AHB) including a rolling bearing, a gas-dynamic multi-foil bearing and a system of the electromagnetic and piezo actuators. The modeling of the rotor dynamics is based on solution of the Lagrange equations of type II, the force factors of the rolling bearing are determined using the Hertz theory, of the gas-dynamic bearing – using the Reynolds equation solution. The mechanisms of electromagnetic and piezo actuators operation are described by a simulation model based on the equations describing electromagnetism and piezo effect. The principle of AHB operation allows to control stiffness and damping properties of the bearing effectively and thereby reduce the vibrations level. Furthermore the AHB eliminates the gas-dynamic bearing foils wear during the start-up and the stopping mode of the rotor. The results of mathematical modeling in the form of Campbell diagrams, the frequency response of the rotor-bearing system, the stiffness of the AHB as well as the approach to determination of the electromagnetic actuators operation algorithm are presented in the paper.

Keywords: Rotor, Hybrid bearings, Stiffness, Natural frequencies, Active control

1 INTRODUCTION

Reliability of rotating machinery operation is considerably determined by the bearings. For the particular applications the requirements in rotation speed, load capacity, temperature and maximal vibration level are so extreme that neither rolling bearings nor fluid-film bearings could provide the appropriate operation characteristics along the whole range of working conditions. A hybrid bearing consists of a fluid-film bearing and a rolling bearing that makes it able to escape the disadvantages and to keep the advantages of the each mentioned type of bearing. The interest in the hybrid bearings has appeared in the 1960s-1970s [1, 2]. In the 1980s, NASA launched several research programs on hybrid hydrostatic/ball bearings for cryogenic turbopumps [3, 4]. These studies demonstrated the benefits of hybrid bearings and feasibility of their use in turbomachinery applications. There are two fundamentally different configurations of the hybrid bearings. According to the abbreviations in [3] these configurations are called PL (parallel load) and PS (parallel speed). Fig. 1,a shows the latter concept of a hybrid bearing with speed separation. In this scheme a fluid-film bearing (FFB) can be mounted on the outer race (PSEX bearing) of the rolling bearing (RB). Such configuration allows to reduce the rotation speed of the rolling bearing to the value smaller than the value of the rotor speed [5]. A shaft rotates in the rolling bearing during the start-up and the stopping mode. A fluid-film bearing is switched on at the main operation mode. As rotation speed (and so hydrostatic pressure) rises, the load capacity of the fluid-film bearing increases and the rolling bearing becomes partially switched off. Thus the lifetime expectancy of the RB is improved due to the smaller DN value that is the product of multiplication of the rotation speed and the shaft diameter [5].
The PL concept shown in Fig. 1,b is called a hybrid bearing with load separation. The rolling bearing and the fluid-film bearing are combined here in a series way. Though the shaft permanently rotates in the rolling bearing, the lifetime expectancy of the RB increases due to the smaller loads at high speeds. The rolling bearing carries the external load during the start-up and the stopping mode. At the main operating mode the fluid-film bearing takes a certain part of the load from the rolling bearing due to the hydrodynamic reaction of the fluid film. Thus the lifetime expectancy limiting load acting on the RB is reduced. Early works on the PL hybrid bearings dealt mainly with the lifetime expectancy improving.

2 APPLICATION OF HYBRID BEARINGS

A literature survey has shown that the attempts to use hybrid bearings were made when neither RB technology nor FFB technology were available and so there was no possibility to provide the necessary reliability and efficiency. The ceramic ball bearing technology put the development of the hybrid bearings aside. But recently the interest in hybrid bearings has appeared again. The aim of this section is to provide a brief overview of the known works on the hybrid bearings. Though the various applications of the hybrid bearings in high-speed rotating machinery systems and heavy-loaded presses are known, in the present paper we will consider only applications for the turbomachinery.

Figure 2 shows the examples of several hybrid bearings constructions. The first group of researches on the hybrid bearings is dated between 1970-1976 (e.g. [3-8]). Most of these studies consider the PS type bearings (series-hybrid bearings). The presented theoretical and experimental results show the lifetime expectancy increase and smaller friction losses due to the reduction of the rolling bearing speed. The France patent on PSEX bearing obtained by SKF is shown in Figure 2, a. A PL bearing with hydrostatic support for an automotive gas turbine is considered in [7] (Figure 2, b). Wearless gas-dynamic bearing operation mode was achieved by using a small rolling bearing which carries the load during the start-up and the stopping mode. In the 1980s, NASA launched several projects on high bearings with the high lifetime expectancy for rocket engine turbopumps. This resulted in the more detailed studies on both PS and PL bearings (see the examples in [3],[4],[8]). A LH2 turbopump was modified with a PSEX bearing to improve lifetime expectancy and dynamic characteristics (Figure 2,c). In [3], all the main configurations of hybrid bearings for turbopumps were investigated to achieve high stiffness and damping characteristics.
Application of hybrid bearings in aircraft engines was studied in the USSR [9]. Figure 2, d shows a PSIN bearing. After that the studies on hybrid bearings were apparently suspended, although new designs were patented, e.g. [10]. Figure 2, e shows the USA patent on the hybrid bearing with the intermediate element dated 1994. The support ring is locked on the shaft with the rolling bearing during the transient process. At the main operation mode the ring bends aside and the load starts being carried by the hydrostatic bearing. Figure 2, f shows the Russian Federation patent on the hybrid foil bearing [11]. Further hybrid bearing configurations which combine PL and PS bearings are also known. Recently the theoretical and experimental investigations have been resumed in Russia [12], [13]. The results of the theoretical and experimental researches on the basic types of hybrid bearings that provide high possibilities for improving the lifetime expectancy, dynamic characteristics and friction reduction are presented in [12,13].

3 THE CONCEPTION OF THE ACTIVE HYBRID BEARING

Developing intellectual control systems for rotating machinery is very laborious from the viewpoint of complicity of algorithms, mathematics, software and practical implementation as well. The potentiality of such intellectual control systems allows to make rational control decisions under lack of information or internal and external conditions uncertainty in terms of providing the machine’s workability. In prospect this type of control can be used for providing stability of ultra high-speed rotors motion with rotation speed about 100000 rpm and more. The controlled parameters that provide the required dynamic and integral characteristics in active hybrid bearings (AHB) are: the gap between the bearing and shaft surfaces; lubricant pressure at the input or at the working area; lubricant flow rate; displacement and velocity of the shaft center; stiffness and damping rates of bearings, seals and dampers; the viscosity and concentration of substances the lubricant; the bearing’s functional structure. The values of these parameters can be adjusted with actuators based on piezo-, electromagnetic, magnetic-hydrodynamic effects; pneumatic, hydraulic and electromechanical actuators.

The terms “hybrid bearings” and “combined bearings” are both commonly used in Russian scientific literature on rotor dynamics and hydrodynamic lubrication theory while non-Russian
authors usually use only the term “hybrid bearings”. There is a certain contradiction in using these terms. The matter is that in some cases it comes to the hydro-static-dynamic or angular contact bearings, i.e. bearings with a hybrid load capacity or the direction of the perceived load; in other cases it comes to combining the different types of bearings and other elements of rotor systems (seals, dampers, vibration absorbers). In the present project the term “combined bearing” is used for designation of a bearing that consists of elements of providing the constant rotor position relating to the housing, motion stability, vibrations level reduction and sealing the unit’s cavities. The generalized structural scheme of a combined bearing is shown in the figure 3, it includes the shaft 1, the housing 2, the rolling bearing 3, the fluid-film bearing 4, the thrust rolling or sleeve bearing 5, the damper 6, the sealing 7, the switching element 8, the control system 9. Proceeding from the combinatorial possibilities the following types of constructions can be designed.

![Combined Bearing Diagram](image)

1 – the shaft, 2 – the housing, 3,4 – the rolling / fluid-film bearing, 5 – the thrust bearing, 6 – the damper, 7 – the sealing, 8 – the switching element, 9 – the control system.

Figure 3. The generalized structural scheme of a combined bearing

The proposed design of the AHB is illustrated in the figure 4: the hybrid bearing consists of the housing 1, rolling bearing 2 mounted in the housing. The sleeve 3 with plates 4 is mounted in the rolling bearing 2. The plates provide centering of the shaft 5. The electromagnetic coils 6 are mounted round in the housing 1. Piezo actuators 7 mounted on the ring 8 are used for locking the inner ring during the main operation mode.

The AHB operates in the following way: at the initial time the load transmission from the shaft 5 to the housing 1 is implemented through the multi-leaf foil bearings 3,4, rolling bearing 2 and the electromagnetic field which is created by the electromagnets 6. With increase of the shaft’s rotational speed the voltage supplied to the magnets decreases, the multi-leaf foil bearings bends off the shaft surface. The gas-dynamic force appears in the gap between the foils and the shaft. The gas-dynamic force centers the shaft and carries the external load. Simultaneously the piezo actuators start operating and rotation of the inner race of the rolling bearing stops. When the rotation speed decreases the whole process is reversed.

This principle of an AHB operation limits the possibility of choosing the material of the multi-wedge bearings foils material. The foils’ material must have diamagnetic properties. In the case of using a ferromagnetic material the principle of an AHB operation changes. At the initial time the load is transmitted from the shaft 5 to the housing 1 through the multi-foil bearing 3,4 and rolling elements of the rolling bearing 2. The electromagnets 6 are switched off and the foils 4 are tightly pressed to the shaft’s surface due to the preliminary assembly deformation. As the shaft’s rotation speed increases the voltage is supplied to the electromagnets 6, the foils 4 move...
away from the shaft’s surface and an air gap appears between them. The gas-dynamic force that centers the shaft and carries the load appears in the gap. Simultaneously the piezo actuators start operating and rotation of the inner race of the rolling bearing stops.

1 – housing, 2 – rolling bearing, 3 – sleeve, 4 – metal plates, 5 – shaft, 6 – electromagnetic coils, 7 – piezo actuators, 8 – ring.

Figure 4. The hybrid bearing.

The control system does not affect the self-regulating properties of the gas foil bearing because the deformation of the elastic foil occurs under the influence of both electromagnetic and gas-dynamic forces produced by the air film. Decrease of the lubricant film load capacity is connected with the radial gap decrease that results in increasing film reaction forces. This process also leads to additional foil deformation, increase of the gap and decrease of the reaction force, thereby the self-regulation of the rotor position is implemented.

4 MATHEMATICAL MODELING

The described device can improve the reliability and the durability of a bearing by separating and duplicating the rolling and fluid-film bearings’ functions and control their characteristics at the different operating modes.

4.1 Multi-foil bearings

Lets present the model of the foil as a cylindrical shell and consider it in $z$, $\theta$ coordinates. The foil deformation under a joint action of electromagnetic and hydrodynamic forces can be determined by solving the differential equations system for a cylindrical shell which origins from the general theory of shells [14].

In (1) $\delta$, $u$, $w$ are foil displacements; $r$ is the curvature radius of a foil; $\delta$ is the foil thickness; $p(\theta,z)$ is the hydrodynamic force; $F_{em}$ is the electromagnetic force; $A$ is the surface area of a foil. The determination of a gas-dynamic reaction force created by one foil is based on the solution of the Reynolds equation, modified for the case of the stationary turbulent two-dimensional flow of a viscous compressible lubricant. The Reynolds equation is derived from the Navier-
Stocks equations system using the qualitative assessments assuming that the gap between the friction surfaces is small comparing to the other dimensions [16].

\[
\frac{\partial^2 u}{\partial z^2} + \frac{1 - \nu}{2r^2} \frac{\partial^2 u}{\partial \theta^2} + \frac{1 + \nu}{2r} \frac{\partial^2 \vartheta}{\partial z \partial \theta} + \frac{\vartheta}{r} \frac{\partial w}{\partial z} = 0; \\
\frac{1 + \nu}{2r} \frac{\partial^2 u}{\partial z \partial \theta} + \frac{1}{r \theta^2} \frac{\partial^2 \vartheta}{\partial \theta^2} + \frac{1 - \nu}{2} \frac{\partial^2 \vartheta}{\partial z^2} + \frac{1}{r \theta} \frac{\partial w}{\partial \theta} = - \frac{1 - \nu^2}{E \delta} \left[ p(\theta, z) + \frac{F_{em}}{A} \right].
\]

(1)

Usage of the Reynolds equation is conventional in the gas lubrication theory and the results obtained with such a mathematical model comply with experimental data well. Also the inertia forces are small comparing to the pressure and viscosity forces, appearing when the air lubricant moves in the gap. Assuming that \( \xi = r \theta \), \( \omega = r \dot{\theta} \), let’s display the Reynolds equation in the cylindrical coordinates:

\[
\frac{1}{r^2} \frac{\partial}{\partial \theta} \left( \rho h^3 \frac{\partial p}{\partial \theta} \right) + \frac{\partial}{\partial z} \left( \rho h^3 \frac{\partial p}{\partial z} \right) = 6 \frac{\partial (\rho \cdot U \cdot h)}{\partial \theta}
\]

(2)

The turbulence coefficients \( K_\delta \) and \( K_z \) allow to take the impact of the additional turbulent viscosity into consideration. The turbulence coefficients can be determined according to [17]:

\[
K_\delta = 1 + 0.044 \left( k' \cdot Re \right)^{0.725}; \\
K_z = 1 + 0.0247 \left( k' \cdot Re \right)^{0.65},
\]

(3)

where \( k' \) is the Carman coefficient, \( k' = 0.2...0.4 \). The small values of the Carman coefficient correspond to the small values of the radial gap \( h = 10...100 \mu m \). The Carman coefficient is often calculated according to the empirical expression \( k' = 0.125 \cdot Re^{0.07} \).

The radial gap function is determined as the difference between the points of the rotor surface and the non-pressed foil

\[
h(n) = \frac{h_0 - w}{N} (N - n) + w - Y_0 \cos(\phi) - X_0 \sin(\phi),
\]

(4)

where \( h_0 \) is a radial gap between the rotor and the place of foil fixing; \( w \) is a foil displacement, \( N \) is a number of elements the foil is divided into; \( n=0 \ldots N \) is a number of the element; \( X_0 \) is the shaft displacement along the \( x \) axis; \( Y_0 \) is the shaft displacement along the \( y \) axis; \( \phi \) is an angle of the relative rotor position in the bearing gap.

Determining the pressure distribution field \( p(\theta, z) \) at a particular time is a boundary value problem of Reynolds equation solution. The Reynolds equation is an elliptic type nonlinear partial differential equation with variable coefficients. For the considered task it is solved with the following boundary conditions:

– the predetermined outflow pressure \( p_a \) on the bearing ends is:

\[
p(r \theta, 0) = p_a; \quad p(r \theta, L) = p_a
\]

(5)

– the bearing surface of the rotor is fully covered with the lubricant film (the Sommerfeld hypothesis).

\[
p(\theta, z) = p(2\pi, z); \quad \frac{\partial p}{\partial r}(\theta, z) = \frac{\partial p}{\partial r}(2\pi, z).
\]

(6)

The reasonableness of such approach is proved, for instance, in [18].
The impact of the lubricant film can be taken into consideration by means of including the hydrodynamic force $R$ and the friction force $F_{fr}$ into the calculation scheme. The projections of the hydrodynamic force on the axes of the moveable coordinate system $IOJ$ linked to a certain foil can be determined from the relevant geometric considerations and the following formulas:

$$
R_j = -\int_0^L p \cdot \sin(\theta - \varphi) \cdot rd\theta dz \\
R_i = -\int_0^L p \cdot \cos(\theta - \varphi) \cdot rd\theta dz
$$

(7)

The bearing capacity $W$ and its direction (the angle $\phi_w$) are determined as follows:

$$
W = \sqrt{R_j^2 + R_i^2} ,
$$

(8)

$$
\phi_w = \arctg \left( \frac{R_i}{R_j} \right).
$$

(9)

The total bearing capacity $W$ of the gas-dynamic bearing is determined by the vector summation of reactions provided by each foil.

The dynamic characteristics of the FFB are determined assuming the linear force-displacement-velocity relations:

$$
R_{f}^x = -K_{xx} \Delta x - K_{yy} \Delta y - B_{xx} \Delta \dot{x} - B_{yy} \Delta \dot{y} ,
R_{f}^y = -K_{xy} \Delta x - K_{yx} \Delta y - B_{xy} \Delta \dot{x} - B_{yx} \Delta \dot{y} .
$$

(10)

The coefficients $K_{xx}, K_{yy},..., B_{xy}, B_{yx}$ are the partial derivatives calculated at the equilibrium position:

$$
K_{mn} = \frac{\partial R_m}{\partial n} \Big|_0 ,
B_{mn} = \frac{\partial R_m}{\partial \theta} \Big|_0 ,
$$

(11)

4.2 Rolling bearings

The classical RB models are based on the Hertz theory. A comprehensive analysis of the rolling bearing can be found in [18], [19]. The rolling elements are deformed under the total load by the $\delta$ value. The force-displacement characteristic of the rolling bearing at the contact point can be expressed with the load-deflection factor $K'$ which is a total stiffness of the outer and inner raceways:

$$
R_f = K' \delta^{3/2} ,
K' = \frac{K'_{i} \cdot K'_{o}}{K'_{i} + K'_{o}}
$$

(12)

It is convenient to rearrange the force-displacement equation (13) in terms of quasi-static stiffness for dynamic analysis:

$$
R_{f}^x = K'_{x} X ,
R_{f}^y = K'_{y} Y ,
$$

$$
\begin{bmatrix}
K'_{x} \\
K'_{y}
\end{bmatrix} = \frac{2}{3} (K')^{2/3} \left[ F_{\Sigma} \left\{ \cos \gamma \right\}^{1/3} \right]
$$

(13)

The equivalent load acting on the RB is summed up from the applied load, the rotor weight and the reaction of the fluid-film bearing acting at the angle $\beta$:

$$
F_{\Sigma} = F_r + mg \cos \gamma - R_f \cos(\gamma - \beta).
$$

(14)

4.3 Electromagnetic actuators

The electromagnetic force acting on a foil is determined according to [15].
\[
F_{\text{res}} = \frac{U}{H} \frac{k_{Cu}}{128\rho\mu_0} \left( \frac{D}{2} - \frac{d}{2} - t \right)^2 \left( \frac{\pi d}{8} - t \right)^2.
\]  

(15)

where \(U\) is the voltage on the coils; \(H\) is the gap between the coils and the foils; \(k_{Cu}\) is the coefficient of filling of the groove \(A\) with copper \((k_{Cu} = 0.3..0.5)\); \(a\) is the coefficient of the poles number; \(\alpha = 0.924\) for the number of poles \(p = 8\) \([15]\); \(lm\) is the average coil conductor length; \(\rho = 0.018\ \text{Ohm-mm}^2/\text{m}\) is the copper resistivity; \(\mu_0 = 4\pi \times 10^{-7}\ \text{H/m}\) is the magnetic constant; \(D\) is the outer coils diameter; \(d\) is the inner coils diameter; \(t\) is the pole width.

Using the equation (2) we can determine the electromagnetic force acting on a foil. It allows to calculate the additional deformation caused by the electromagnetic forces. These forces can be adjusted by changing the voltage on the coils and thus the foil displacement control can be implemented.

### 4.4 Multi-mass rigid rotor

The dynamic behavior of a rotor on the AHB can be studied by means of the mathematical model of a multi-mass rigid asymmetrical rotor. The scheme illustrating the model is shown in Figure 5.

\(G\) – the center of mass, \(\xi\) – the principle central rotor axis, \(\delta\) – the angle between the rotor symmetry axis and the principal central axis, \(e\) – the rotor imbalance

**Figure 5.** The model of the rigid asymmetrical rotor

The equations of the rigid asymmetrical rotor motion were obtained using the Lagrange's equations \([21]\)

\[
\frac{d}{dt} \frac{\partial T}{\partial q_i} - \frac{\partial T}{\partial q_i} = \Sigma W_i ,
\]

(16)

where \(q_i\) is the generalized coordinate, \(W_i\) is the generalized force at the \(i\)-th movement.

According to the Kenig’s theorem the rotor’s kinetic energy is:

\[
T = \frac{1}{2} m \left( \dot{x}^2 + \dot{y}^2 \right) + \frac{1}{2} I_z \omega_z^2 + \frac{1}{2} I_x \omega_x^2 + \frac{1}{2} I_y \omega_y^2.
\]

(17)

where \(m\) – is the rotor mass, \(\omega_z, \omega_x, \omega_y\) are the projections of an angle velocity vector on the principal central coordinate axes, \(I_z, I_x, I_y\) are the principal inertia moments.

The generalized forces acting on the rotor are:
The transformation of the equations (17) gives the equations describing the dynamics of the multi-mass rigid asymmetric rotor:

\[
\begin{align*}
\Sigma W_1 &= \frac{1}{Z_2 - Z_1} \sum_{k} M_k + \frac{1}{Z_2 - Z_1} \sum_{m} F_m \left( Z_2 - Z^F_1 \right), \\
\Sigma W_2 &= -\frac{1}{Z_2 - Z_1} \sum_{k} M_k + \frac{1}{Z_2 - Z_1} \sum_{m} F_m \left( Z^F_m - Z_1 \right), \\
\end{align*}
\]  

(18)

The computational experiment was carried out with the rotor mounted in the oxygen turbopump with rotation frequency of 83000 rpm. The AHB for the turbopump is being developed by the paper’s authors under the agreement on scientific cooperation with JSC «Konstruktorskoe Buro Khimavtomatiky» [22]. The following parameters were used in the calculations: the Young’s modulus is \(2 \times 10^{11} \text{ Pa}\), the Poisson’s ration is 0.3; the density is 7850 kg/m\(^3\); the full mass of the rotor is 4.3 kg; the rotor inertia moment is \(4.88 \times 10^{-3} \text{ kg} \cdot \text{m}^2\); the rotation speed is 83000 rpm.

5 \text{ ACTIVE HYBRID BEARINGS PERFORMANCE}

Active control allows to change the stiffness and damping properties of an AHB, that helps to change the natural frequencies spectrum of the rotor system. Figure 6 shows the Campbell Diagram for a rotor system with AHB. The point I corresponds to the first possible state of the rotor system when the electromagnets are switched off and the stiffness of the bearing is about \(2.2 \times 10^{8} \text{ N/m}\). The point II corresponds to the second possible state when the electromagnets are switched on and the stiffness of the bearing is about \(3.7 \times 10^{10} \text{ N/m}\). When approaching the critical frequency which corresponds to the first state of the rotor system, the value of the
electromagnets supply voltage changes. The rigidity of the AHB decreases and the rotor system moves to the second state. The second state is corresponded to the value of the critical frequency which is less than the actual rotor speed. When reaching the operating speed the control parameters can be adjusted for ensuring the minimum vibrations level in the system.

Figure 6. Campbell diagram for a rotor on the AHB.

Figure 7a shows the results of the theoretical calculation of the AHB stiffness coefficients at the different operation modes depending on the rotation speed. Figure 7b shows the frequency response of the rotor on AHB.

Figure 7. The AHB’s stiffness depending on the rotation speed (a) and the frequency response of a rotor for two stiffness levels (b)
The figure shows the process of switching between the two AHB’s stiffness levels and its influence on the value of vibrations amplitude. The rotor has 2 different natural frequencies for the each value of the AHB’s stiffness levels. Lets formulate the control task as overpassing the resonance frequencies with the minimal value of vibrations amplitude. Then the control process will be as follows. During the acceleration the rotor approaches to the first critical frequency for the first AHB’s stiffness level. At this moment the bearing gets switched to the second stiffness level where the value of vibrations amplitude is of the range after the first critical frequency of the second stiffness level. Approaching to the second critical frequency of the second stiffness level causes the back switching to the first stiffness state, so the value of the vibrations amplitude becomes of the range after the second critical frequency of the first stiffness level. This algorithm completely eliminates the possibility of increasing the value of vibrations amplitude when passing the critical frequencies. The algorithm can also be extended to passing the following critical frequencies before the nominal rotation speed of the rotor. During the deceleration the processes go backwards.

Knowing the critical frequencies values (figure 8a) and the acceleration characteristic of a rotor machine (figure 8c) a control algorithm for the electromagnetic actuators can be developed (figure 8b).

![Figure 8. The connection between the dynamic parameters of the rotor system and the control voltage at the actuators](image)

The theoretical model of a multi-mass rigid rotor on the AHB allows to determine the frequency response of the rotor, reveal the resonance zones range considering the variable stiffness of the bearings, determine the mutual influence of the combined bearings’ force factors and the necessary time of the electromagnetic actuators stay in an ON or OFF state in order to reduce the value of the vibrations amplitude when passing the resonance frequencies.

6 ACKNOWLEDGEMENTS

The work has been carried out within the project No. 14.574.21.0044 "Development of Active Hybrid Bearings for Aircraft Rotating Machinery" supported by the federal target program “Research and development on the priority directions of the scientific-technological complex of Russia for 2014-2020” by the Ministry of Education and Science of Russian Federation (unique identifier research RFMEFI 57414X0044).
REFERENCES


Heterogeneous, multi-tier wheel ground contact simulation for planetary exploration

Roy Lichtenheldt, Matthias Hellerer, Stefan Barthelmes, Fabian Buse

Deutsches Zentrum für Luft- und Raumfahrt / German Aerospace Center (DLR)
Robotics and Mechatronics Center
Institute of System Dynamics and Control
Münchener Straße 20, 82234 Weßling, Germany
Roy.Lichtenheldt@DLR.de / Matthias.Hellerer@DLR.de

ABSTRACT

Today’s growing scientific interest in extraterrestrial bodies increases the necessity of extended mobility on these bodies. Thus, planetary exploration systems are facing new challenges in terms of mission planning, obstacle and soil traversability. In order to fit the tight schedules of space missions and to cover a large variety of environmental conditions, experimental test setups are complemented by numerical simulation models used as virtual prototypes. In this context we present an integrated simulation environment which allows for using different available contact models, ranging from simple but real-time capable approximations based on rigid-body modeling techniques up to very accurate solutions based on Discrete Element Method (DEM). For this work, a one-point Bekker based approach and the so-called Soil Contact Model (SCM), which is a multi-point extension of the BEKKER-WONG method taking soil deformation into account, are used. These two contact models are applied for homogeneous models with only one type of contact model for all wheels as well as for a heterogeneous multi-tiered model with different contact models for the wheels. It will be shown that the multi-tiered approach enhances the simulation result accuracy compared to the results of a homogeneous model with a low level of detail while speeding up the simulation in comparison to a homogeneous higher tier model.

Keywords: Wheel-Ground Interaction, Terramechanics, Soil Contact, Planetary Exploration.

1 INTRODUCTION

In order to further understand the formation of planets and our solar system, planetary science requires extended mobility for the exploration of extraterrestrial bodies. Therefore, the locomotion sub-systems enabling planetary exploration are facing new challenges in terms of durability, mission planning, obstacle and soil traversability. Testing in the actual environmental conditions is often very expensive and time-consuming or not even possible. Additionally, environmental conditions on the site of operation are often uncertain and not well-known beforehand. Thus, in order to fit the tight schedules of space missions and to cover a large variety of environmental conditions, experimental test setups are complemented by numerical simulation models used as virtual prototypes.

In this context we present an integrated simulation environment for the heterogeneous simulation of the wheel ground contact using tiered contact models, specially designed for planetary rovers. These models range from simple but real-time capable approximations based on rigid-body modeling techniques, over penetration and soil deformation approaches, to very accurate but slow particle methods. Having these different techniques available in one environment allows us to directly compare results and to use them in conjunction with each other. We will show how this approach can on one hand be used to improve the simulation accuracy of faster simulation techniques and on the other hand improve the simulation speed of more accurate techniques. The different tiers will be explained in detail, tangible comparisons are given and their respective applications for current planetary exploration missions, as well as their limits are discussed. In this context we also discuss a future verification of our simulation models.
The comparisons of single homogeneous and multi-tiered heterogeneous wheel ground contact will be carried out using our in-house Soil Contact Model (SCM). This previously verified Model is still capable of being used in the efficient computation times of multi-body dynamics and is thus used as the highest tier. In conjunction with SCM, a lower tier, single point contact model is used in order to speed-up full system simulations and thereby to support applications reaching from control design to actual mission planning. The most detailed DEM models are taken into consideration for future enrichment of lower tier models as well as for the usage in the multi-tier environment. The particle-based models are still found to be too computational expensive in order to efficiently use them in large scale, full system simulation. Hence, this article is aimed to be a first feasibility study for multi-tiered wheel-ground contact, as well as to exemplify its advantages and raise questions in this field.

2 THE DLR ROVER SIMULATION TOOLKIT FOR MODELICA

Modelica is a multi-physics, object oriented modeling language. Base objects are defined by equations and interfaces, which are then assembled into larger, more complex objects, which again might be part of an even larger component. In this way very large and complex multi-physics systems, spanning multiple domains, such as mechanical and electrical, can be modeled consistently and simulated subsequently.

The DLR Rover Simulation Toolkit is a Modelica library covering all relevant physical subsystems of a planetary rover, such as drive-trains, sensors and electrical systems, with a special focus on locomotion in general and the wheel-ground contact in particular. Using Modelica allows us to utilize a large number of custom and commercially available libraries. The toolkit is designed to be essentially modular. All parts are replaceable by either different simulation models or HIL systems, with a standardized interface. Thus, it provides an ideal environment for the following investigations. The structure and adaptability of the framework is illustrated in Fig 1, showing the setup with focus on ground contact simulation component. The Mulibody system (MBS) simulates the movement of the objects according to applied torques and forces. On the one hand this system is influenced by the simulation of the drive-train and attached systems, on the other hand by the ground contact. As shown, the ground contact models all use a common interface to the MBS. The MBS communicates the current position and orientation of the wheels, including derivatives, to the ground contact simulation, which responds with resulting forces and torques.
2.1 Rigid Body Contact
The simplest simulations of multi-body dynamics are typically based on rigid bodies only. Therein a rigid body is defined as an idealized, perfectly non-deformable object, independent of the external forces acting upon it. Connections, joints and contacts always maintain their imposed constraints and hard impacts between objects use impulse transfers to avoid any penetration. While neglecting many effects of real world objects, the results are still sufficiently accurate for many applications. The big advantage of this approach lies in the small required simulation time: with modern desktop computers even complex scenarios can be simulated in real-time, thus this technique is favored in animation and gaming industry [1] [2] [3].

2.2 Penetration-based single point Penalty Models
More accurate results can be achieved by incorporating the deformability of real bodies. Depending on the simulation focus, joints or contact and impact points are modeled by adding virtual spring damper elements. For simple contacts only the penetration depth is taken into account. Especially critical for the real-time capability are multiple such virtual spring damper elements in a serial configuration and stiff contacts, as required for the accurate simulation of many real world materials [3]. The rigid body and penetration-based approaches are compared in Figure 2. Both approaches feature single point contact detection between a contact sphere of constant radius and a CAD surface, but can be adapted to surface-surface contacts. Effects like grouser-induced effects and soil displacement are not covered by these models, as they are not directly dedicated to terramechanics. Thus, they are mainly used for real-time applications and first estimates in early mission phases. Furthermore, the rigid body approaches are suitable for the simulation of hard contacts between wheels and stones. In order to cover the forces and torques, as well as the sinkage behaviour of locomotion in soft, sandy soils, the well known BEKKER-WONG theory is applied in our framework. Therefore, several contact points can be taken into account to cover the contact between cylindrical wheels and non-convex surfaces. However the forces and torques are calculated with a single evaluation of the contact equations by averaging the position as well as the contact normal over all contact points. The extended BEKKER-WONG terramechanics model for the simulation of off-road locomotion is parametrized by seven parameters. These val-
ues characterize a specific soil and can be measured empirically using a specialized device called 'Bevameter'. We performed many empirical tests and possessed an extensive collection of parameter sets for different soils. The model is best known for its role in the design of the Lunar Roving Vehicle [4], but has found popularity in the development of planetary exploration rovers. The model doesn’t immediately prompt an implementation, but additional assumptions are necessary for a numerical simulation. Our implementation is based on the pressure-sinkage relationship, proposed by Bekker as follows:

\[ p = \left( \frac{k_c}{b} + k_\phi \right) z^n \]

(\( p \): pressure, \( z \): sinkage \( b \): smaller dimension of contact patch, \( n, k_c, k_\phi \): Bekker’s soil parameters [5])

This may then be used in the calculation of the shear stress according to Mohr-Coulomb with extensions by Janosi-Hanamoto:

\[ \tau = \tau_{\text{max}} (1 - e^{-j/K}) = (c + \sigma \tan \phi) (1 - e^{-j/K}); \quad \sigma \equiv p \]

(\( \sigma \): normal stress, \( \tau \): shear stress, \( \phi \): angle of repose, \( j \): shear displacement, \( c \): cohesion, \( K \): shear deformation parameter)

Finally a multi-body simulation is used to determine the contact area between the locomotion system and the ground. Integrating the normal and shear stress over the contact area yields the resulting contact forces [6] [5]. Furthermore, we use a custom extension of this model to incorporate the effects of grousers based on Rankine’s passive earth pressure [7]. However the soil displacement by the wheel is not covered.

The model is suitable wherever fast simulations covering the basic effects of terramechanics are required.

2.3 Soil Contact Model - SCM

Up to this point the soil did not actually deform. However one of the most important effects for the simulation of planetary rovers is the plastic deformation of soil caused by the wheels. In our simulation framework we use the SCM (Soil Contact Model) algorithm for the simulation of soft soil contact forces and plastic deformation of the soil. SCM is a in-house developed, highly specialized, three dimensional, novel extension of the well known Bekker-Wong method [8, 9]. The main features of SCM are:

- Surface contact between arbitrarily shaped objects,
- Soil forces and sinkage calculated for every discretization node based on the Bekker-Wong theory,
- Plastic soil deformation covered by deformable surfaces,
- Soil displacement by empirical flow field and erosion algorithm.

Analogue to the previously introduced models, SCM is based on surface contacts for both wheel and soil. In contrast to the single point contact models, which use the soil surface only for contact detection, SCM calculates the soil deformation and reaction forces at every single node in the contact patch. Therefore, only the nodes of the wheel grid are used to form a discrete Bekker theory model.

The sinkage and normal pressure are calculated based on Bekker’s theory [5] by weighting the pressure distribution based on the centrality distribution over the contact area [9]. In order to cover plastic soil deformation and thus rutting, as visualized in Figure 3, the portion of soil within the footprint is disposed to the nearest-neighbor nodes outside the contact area. Thereafter the displaced soil is settled by an erosion algorithm iteratively. Thus, the maximum angle of repose as
well as landslides induced around the wheels can be covered by SCM’s plastic soil deformation. Using this approach, SCM enables to cover the main effects of terramechanics and soil deformation, namely bulldozing, rutting and multipass, in the environment of multi-body dynamics in an efficient way. Therein multipass and rutting are covered mainly geometrically. While the volume of disposed soil and its strength are influenced by a plasticity parameter, the soil parameters themselves remain unchanged. SCM has been successfully used in the simulation of planetary rovers [9] and the evaluation of its control using multi-body dynamics [10]. A first verification of the model was carried out in [9] for models of pressure sinkage tests, as well as full-system scale tests. Further validation is currently performed using the DLR-RMC single wheel test facility.

### 2.4 Discrete Element Method - DEMETRIA

The most detailed models are based on particle methods, i.e. the Discrete Element Method (DEM). These methods allow to model regolith directly as granular material without the need of empirical relations. However even for modern powerful computers, simulations using the real grain size are still not possible. Thus, the DLR-SR particle dynamics framework "DEMETRIA" (Discrete Element Method Enabled Terramechanics Interaction framework), based on the particle simulator Pasimodo, is incooperating systematic particle scaling and a priori parameter estimation. The framework’s main features and advantages are explained in [12] and [13]. DEMETRIA was successfully applied to simulation of planetary rover wheels, exemplifying wheel-optimization potential [11]. Additionally, it was applied to InSight’s [14] subsurface locomotion system: The HP3-Mole [15] was simulated using co-simulation of particle and the MBS mechanism model [13] and influences of the outer shape on the performance were shown [16]. The particle-based soil models have been verified and validated using several kinds of material tests, usually used for characterization of soils. In addition to that the HP3-Mole’s co-simulation results are validated against deep penetration tests. The DEM wheel models have been checked for their qualitative behaviour in worst-case soils as exemplified in Figure 4 and are currently being validated using the DLR-RMC single wheel test facility. However due to the high demand on computation time and power, DEM is not suitable for the
simulations of long trajectories at full vehicle level. Thus, these models are mainly used in order to investigate and understand the low-level effects of the interaction and thereby to enrich more efficient models. Hence the particle-based models will not be used for further investigations in this article, but are considered for future heterogeneous wheel-ground contact studies.

3 MULTI-TIERED WHEEL-GROUND CONTACT

In this section it will be shown, how the availability of diverse simulation techniques in a unified environment can be exploited to improve both the simulation speed and results. In order to characterize the soil, Bekker’s parameters derived for the DLR-RMC soil RMCS-13 using the DLR-RMC Bevameter and identification approach [17] will be used. The used simulant is based on calcium carbonate, which represents a worst-case soft soil for wheeled rovers, currently used in test campaigns at DLR-RMC. Utilized in a loose configuration the simulant is almost impassable due to excessive sinkage. Nevertheless the soil is highly compressible and can be prepared in different conditions in order to represent different kinds of planetary regoliths.

3.1 Approach

Coupling fast and slow models can drastically improve simulation times compared to homogeneous higher tier models with admissible influence on the results. Therefore, each wheel’s contact model needs to cover the main effects of the interaction. The leading wheels of a planetary rover are usually driving through loose, uncompacted soils. Thus, their model needs to not only cover the current sinkage and reaction forces, but also the soil displacement causing additional resistance due to bulldozing, as well as the generation of ruts. These ruts will lower the trailing wheel’s driving resistance and at the same time apply higher lateral guidance forces. In order to cover the rutting, SCM is used for the rover’s leading wheels, whereas the trailing wheels are modeled one tier lower as single point Bekker in order to study the approach’s feasibility. The other models presented in section 2 will not be used for further investigation in this article, but are considered for future research in the field of heterogeneous wheel-ground contact. Figure 5 shows the associated effects of the proposed heterogeneous multi-tier contact using SCM’s deformed soil for the contact detection of the trailing wheels. Thereby the Bekker simulated wheel’s results are enhanced by the trailing SCM wheel’s presence, such that the overall motion behaviour is reaches a similar accuracy as the full rover using only SCM. The influence of the used soil simulant’s compressibility on the soil parameters for the trailing wheels is neglected in this first study. Thus, the same soil parameters are used for both SCM and Bekker contact model.

In addition to this approach, simpler wheel-soil models can be improved by offline simulations of higher tier models, such as DEM. By gaining a better understanding of the underlying processes, many of which would be extremely hard or even impossible to analyze in real world tests. As an example SCM’s soil deformation can be checked and enhanced by the farther detailed DEM models.

![Figure 5. Multi-tiered heterogeneous wheel soil contact](image-url)
3.2 Virtual Test Setup

In order to compare homo- and heterogeneous ground contact simulations, a simulation scenario for the evaluation of the tractive performance of planetary rovers is used. The rover is driving along a linear trajectory in a plane of soft soil, sufficiently long to reach a quasi-stationary state of the wheel ground interaction. Thereafter the rover is facing a $20^\circ$ slope and climbs upwards, ensuring that a stationary state is reached again. This scenario was chosen, as the $20^\circ$ sloped terrain is a common test setup for the locomotion capabilities of planetary rovers. The rover itself is modeled as a single track system with two wheels, modeling the symmetry of a four wheeled rover. In Table 1 the parameters of the simulation domain and the full rover are given. In addition to these parameters, the BEKKER model features parameters, which are not only influenced by the soil condition, but also by the soil’s load history and the actual dynamics of the wheel. As there is no widely accepted method to choose these parameters, their identification is done based on empirical knowledge gathered from measurements. In order to check the consistency and the applicability as well as the potential speed up of the heterogeneous multi-tiered wheel ground contact, each rover a homogeneous simulation for each contact model is performed first. Therein it can also be checked whether the usage of one higher tiered contact significantly improves the simulation accuracy of the large-scale models. Figure 6 shows the single-tiered homogeneous, as well as the multi-tiered, heterogeneous setups. The different setups are compared using the following objectives:

- real-time factor $r_{rt}$
- tractive forces $F_y$ and torques $M_z$
- quasi-stationary translational velocity $v_{abs}$
- sinkage $s_{zf}$ of the individual wheels

To allow for improved comparability, the results will be normalized using SCM’s results as a reference. Thereby normalized values are noted as $(..)_0$ and allow to directly compare the signals in terms of their deviations. The sinkage is calculated as absolute sinkage for all wheels. Explicit EULER integration scheme is utilized for all simulations carried out in this work. The solver’s time step size is kept constant at the highest possible, but lowest required for all the models in order to omit influences on the simulation speed and accuracy due to the solver’s step size control.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>wheel diameter $d_w$</td>
<td>250 mm</td>
</tr>
<tr>
<td>wheel width $b_w$</td>
<td>125 mm</td>
</tr>
<tr>
<td>grouser height $h_{Gr}$</td>
<td>10 mm</td>
</tr>
<tr>
<td>number of grousers $n_{Gr}$</td>
<td>12</td>
</tr>
<tr>
<td>full rover mass $m$</td>
<td>100 kg</td>
</tr>
<tr>
<td>angular velocity of the wheel $\omega_w$</td>
<td>1 rad/s</td>
</tr>
<tr>
<td>total size of the domain</td>
<td>$4000 \times 400 \times 728$ mm$^3$</td>
</tr>
<tr>
<td>$k_c$</td>
<td>$7075 \frac{N}{m}$</td>
</tr>
<tr>
<td>$k_\phi$</td>
<td>$29850 \frac{N}{m}$</td>
</tr>
<tr>
<td>$n$</td>
<td>0.83</td>
</tr>
<tr>
<td>angle of internal friction $\phi$</td>
<td>$38^\circ$</td>
</tr>
</tbody>
</table>

4 RESULTS

In the following section we will discuss the impacts of the proposed multi-tiered technique on both the accuracy of the simulation results with respect to the higher, single-tiered model, as well as the
Figure 6. Virtual testscenario to evaluate the tractive performance of different rovers: SCM-SCM (left), BEKKER contact-BEKKER contact (middle), Heterogeneous SCM-BEKKER contact (right)

demand in computation time.
The tractive forces and torques as well as the wheel load in z-direction are used as respective results to compare the accuracy. Together with the individual wheel sinkages and the effective translational speed of the wheel, these values are usual outputs of tractive performance tests for planetary rovers.
The computational performance is rated by the real-time factor which is defined as
\[ r_{rt} := \frac{t_{sim}}{t_{cpu}}, \]
whereby \( t_{sim} \) is the simulated time and \( t_{cpu} \) the total used CPU-time.

4.1 Accuracy of the multi-tiered approach

By applying the common approach of single-tiered contacts it can be shown that the wheel loads differ for both homogeneous setups as certain important effects of soil deformation are not covered by the single-tier BEKKER model. Compared to the latter, the heterogeneous wheel ground contact is able to enhance the results for both front and rear wheel. As the leading wheel (Fig.7) in the mixed-tier simulations is identical to the one in the reference model and thus the comparison is trivial, we will mainly focus on the rear wheels’ comparison. By coverage of the soil deformation by the leading SCM-wheel the tractive performance of the multi-tiered model is increased towards the SCM-SCM level. This is especially evident in Figure 8 after the leading wheel reached the slope at \( s_x = 0.95 \text{ m} \) and after the rover reached the stationary state on the slope. Nevertheless the

Figure 7. Loads on the front wheel
results are always closer to the reference as the homogeneous **BEKKER** contact model is. However the approach still shows differences in the dynamic behaviour compared to the reference model. The main differences are marked in Figures 7 and 8 with I-IV. At point I the disparity is caused by the deviation in the sinkage of front and rear wheel. As can be seen in Figure 9 the rear wheel sinkage $s_z$ for the homogeneous **BEKKER** model as well as the heterogeneous approach are at $\approx 67\%$ of the reference model. As both SCM and **BEKKER** contact feature the same soil parameter, the decreased sinkage causes lower forces and torques accordingly. At point II the

![Figure 8. Loads on the rear wheel](image)

difference in the multi-tier behaviour is directly induced by the approach itself. Figure 10 shows the results of the multi-tiered simulation on the top and the reference model on the bottom. In the pure SCM scenario the front wheel generated a rut in the soil, whereby during the initial sinkage process, material is displaced not only to the front and sides of the wheel, but also to the rear. As the trailing SCM wheel approaches the piled soil, the bulldozing force is increased until the wheel drops into the rut of the leading wheel. Using the multi-tiered model, the trailing wheel does not feature soil deformation and thus the piled soil is not displaced. However the displaced soil is used in contact detection and causes increased forces on the trailing wheel compared to the reference scenario, as well as a lift of the wheel itself (Fig. 9 middle). Due to this lift, the dynamic impact when dropping into the rut is increased as well which causes higher sinkage. Due to differences in sinkage and thus weight distribution, the deviation observed in III and IV occurs. These differences also lead to disparities in rut formation in SCM which in turn causes deviations at the front wheel, too.

### 4.2 Analysis of the computational effort

Regarding the computational performance of the simulation, Figure 11 shows that the homogeneous **BEKKER** model executes faster than real-time. As expected SCM features the lowest real-time factor, as it is the most complex of the models and the only one carrying out several iterations per time-step. In contrast to that the speed-up of the multi-tiered model is not as high as originally expected. While the homogeneous **BEKKER** model yields a real-time factor of 0.52, the multi-tiered model yields $r_{rt} = 5.15$ and the SCM reference model yields in a factor of 6.44. During first investigations on this issue, it was found that the plastically deformable surface supplied by SCM
causes higher computational effort, than the BEKKER model running on a rigid surface. In order to check the influence, the homogeneous BEKKER model was executed on a plastically deformable surface supplied by SCM. It was found that the real-time factors are: \( r_{rt}^{\text{flex}} > 5 \cdot r_{rt}^{\text{rigid}} \). The lower than estimated run-time speed is caused by the synchronized communication between Modelica and the visualization if the SCM supplied surface is used. If the connection is not synchronized the multi-tiered model’s real-time factor decreases to roughly 70% of the above mentioned value, which is within the expected range. Additionally, the homogeneous BEKKER contact model is sped-up as well, but in contrast SCM’s execution time stays constant. However as it enables a more robust contact detection, the synchronization option is kept in favor of the simulation results.

Summing up this first feasibility study for multi-tiered wheel ground contact, it can be stated
that the usage of heterogeneous wheel ground contact simulations improves computational performance compared to full higher-tier simulations on one hand and enhances the level of detail compared to lower tier simulations on the other hand.

As SCM is already verified by previous analysis, it is used as a reference model within this work, however it is still not entirely validated and shows effects that are not compliant with reality for the used type of regolith, e.g. the bulldozing is overpredicted for calcium carbonite based soils. Such effects will be focused on in a currently ongoing in-depth validation campaign of wheel soil models at DLR-SR.

5 CONCLUSION

In the article we showed the integration of wheel ground contact models of different level of detail in a unified simulation framework to allow for the simulation of the various tasks in planetary exploration. Due to the integration of these models in one framework, multi-tiered heterogeneous wheel ground contact modeling is possible in a unified manner.

By usage of our BEKKER and SCM contact model implementations we exemplified this approach in order to achieve a speed up of the simulation while still maintaining nearly the same level of accuracy for the simulation of planetary rover locomotion. Additionally, drawbacks of the approach were shown. In terms of the computation time, the observed speed up was lower than expected. Possible reasons for this were investigated and it was shown that by neglecting synchronization the expected speed-up can be reached.

As calcium carbonite based sands feature excessive compressability, a next step will be the investigation of multi-pass effects in precompressed ruts in compressible simulants. Additionally, further validation of the single models as well as investigations on lowering the computational effort due to the plastically deformable surface will be performed. Moreover in order to allow a deeper insight in the potential speed up, rovers with increasing number of wheels, featuring leading wheels’ SCM contact and lower-tiered contacts for the trailing wheels, will be compared in future work. It is expected that the benefit of the increased accuracy of the soil interaction models is decreasing with a higher number of the multipasses.

Moreover, validated models, e.g. knowledge about deformation fields from DEM can be used to enhance the soil relocation algorithm in SCM.

ACKNOWLEDGEMENT

Parts of this work have been granted by the Helmholtz-Gemeinschaft Deutscher Forschungszentren e.V. under contract number HA-304 (Robotic Exploration of Extreme Environments - ROBEX)

REFERENCES


Numerical Simulation of \( N \)-Body Asteroid Aggregation

Fabio Ferrari\(^*\), Alessandro Tasora\(^#\), Pierangelo Masarati\(^*\), Michèle Lavagna\(^*\)

\(^*\) Department of Aerospace Science and Technology
Politecnico di Milano
Via La Masa 34, 20156 Milano, Italy
[fabio1.ferrari, pierangelo.masarati, michelle.lavagna]@polimi.it

\(^#\) Department of Industrial Engineering
Università degli Studi di Parma
Viale delle Scienze 181/A, 43100 Parma, Italy
alessandro.tasora@unipr.it

ABSTRACT
Recently, the development of dedicated numerical codes has pushed forward the study of \( N \)-body gravitational dynamics leading to better and wider understanding of processes involving the formation of natural bodies in the Solar System. A major branch includes the study of asteroid formation: evidence from recent studies and observations support the idea that small and medium size asteroids between 100 m and 100 km may be gravitational aggregates with no cohesive force other than gravity. This evidence implies that asteroid formation depends on gravitational interactions between different boulders and that asteroid aggregation processes can be naturally modeled with \( N \)-body numerical codes implementing gravitational interactions. This work presents a new implementation of a \( N \)-body numerical solver. The code is based on Chrono::Engine [1]. It handles the contact and collision of large numbers of complex-shaped objects, while simultaneously evaluating the effect of \( N \) to \( N \) gravitational interactions. A special case of study is considered, investigating the relative dynamics between the \( N \) bodies and highlighting favorable conditions for the formation of a stable gravitationally bound aggregate from a cloud of \( N \) boulders. The code is successfully validated for the case of study by comparing relevant results obtained for typical known dynamical scenarios. The outcome of the numerical simulations shows good agreement with theory and observation and suggests the ability of the developed code to predict natural aggregation phenomena.

Keywords: \( N \)-Body Problem, Asteroid Aggregation, Contact Dynamics, Numerical Simulation.

1 INTRODUCTION
Nowadays, small celestial bodies such as asteroids and comets represent the new frontier of the exploration of the Solar System. Space missions aimed at the exploration of these bodies are motivated by great scientific interests, and represent a great challenge for modern space engineering. In particular, the effective design of trajectories to fly a spacecraft in the proximity of an asteroid requires the knowledge of the physical, inertial and dynamical properties of such complex and irregular bodies. Few strategies can be adopted to model to different levels of accuracy the gravity field about asteroids. Classic methods consider a spherical mass distribution of the asteroid and include harmonic expansion of gravitational potential [2] to model the irregularities of the field. Other methods model the asteroid as objects with specific shapes, such as a homogeneous ellipsoids [3] or polyhedra [4, 5]. The accuracy of each method depends on the target body that is modeled and on the application. Typically, each model fits a specific class of asteroids and application range. In the late 1970s, Chapman [6] used the term “rubble pile” to indicate a gravitational aggregate of boulders. Recent studies and observations support the idea that comet nuclei and asteroids between 100 m and 100 km in size may be gravitational aggregates [7]. Such aggregates have very low tensile strength, possessing no cohesive force other than gravity. Asteroids of this class (gravitational aggregates in the followings) can have up to 40% porosity (void fraction). For
this reason, homogeneous mass models of asteroids are usually not suitable to model gravitational aggregates with high accuracy, since internal voids can significantly affect the outer field.

This evidence motivates the present work. The ultimate output of this study is to obtain a high accuracy model of the asteroid mass distribution by studying it as a gravitational aggregate. In detail, the purpose is to study the gravitational aggregation process of a cluster of boulders and to investigate favorable conditions for the formation of a stable aggregate. The problem to be investigated is twofold: (a) the study of gravitational aggregation dynamics, and (b) the study of the physical and dynamical properties of the final aggregate. The first aspect includes the analysis and numerical simulation of typical scenarios, for small and medium-size (hundreds of meters) asteroid aggregation, to identify dynamical conditions that lead to the formation of the aggregate or to the dispersion of the particle cloud.

The physical problem is modeled as a classical $N$-body problem, with mutual gravitational interaction between all particles. Collision detection is implemented and contact forces are included to describe the dynamics of the colliding bodies. The $N$-body problem is a well known mathematical problem, with an established mathematical formulation, but the full comprehension of its solutions and dynamical behavior is still very far from being reached. It has been proved that no analytical solution exists: the problem is characterized by a highly non-linear (chaotic) behavior, which is reflected in a strong dependency of the solution on initial conditions. From the numerical point of view, the $N$-body problem is usually approached using two main classes of codes: $N$-body integrators and $N$-body simulators [8]. Integrators solve the Newtonian equations of motion by computing all $N$ to $N$ gravitational interactions between particles. Simulators incorporate models of dynamical or physical effects to partially estimate the behavior of the $N$ bodies. As a consequence, the high accuracy of numerical integrators is associated with long computational time, whereas simulators are usually faster. The selection of the algorithm is strictly dependent on the application, namely on the physical phenomenon that must be reproduced. Concerning the accretion of planetesimal and asteroid aggregation processes, relevant implementations include tree codes [9, 10, 11], hybrid codes [12], adaptive algorithms of optimal orders [13], systolic algorithms [14] and, more in general, symplectic codes [15, 16, 17]. At the time being, typical capabilities of $N$-body integration software include the handling of few hundreds of bodies with simple (spherical) shape. Collisions and contact interaction between particles are often resolved by interfacing with hydrodynamics codes [18]. The problem is here implemented using Chrono::Engine (C::E in the following) [19], which is able to handle the contact and collision of large numbers of complex-shaped objects.

After this introductory chapter, the dynamic model and its implementation are shown in Section 2. The set up of numerical simulations is presented in Section 3 and the results are discussed in Section 4. Concluding remarks and future work are summarized in Section 5.

2 DYNAMICS AND IMPLEMENTATION

This chapter presents the dynamical model used for the gravitational and contact forces between particles and the numerical implementation of the code using C::E. From a computational standpoint, the problem is characterized by the need to consider long-time simulations of a large number of interacting particles, simultaneously subjected to gravity and contact forces. Gravity implies many-to-many interaction, which depends on the inverse of the distance squared. Contact forces require collision detection and the handling of non-smooth problems. C::E has been selected for its peculiar ability to efficiently and effectively simulate the collision and interaction of large numbers of irregularly-shaped bodies.
2.1 Gravitational dynamics

The classic Newton’s law is implemented to solve for gravitational interactions between the \( N \) bodies, that are here considered as point masses

\[
m_i \ddot{R}_i = G \sum_{j=1, j \neq i}^{N} \frac{m_i m_j}{\|R_{ij}\|^3} R_{ij} \quad \forall i = 1 : N
\]

(1)

with \( R_i \) representing the position vector of body \( i \) in an inertial frame and \( R_{ij} = R_j - R_i \), while \( m_i \) represents its mass and \( G \) is the universal gravitational constant. As shown in Section 3.1, this work investigate the effect of different initial condition sets on the dynamical evolution of a cluster of bodies. Initial conditions are given to initiate the relative distance and velocity between bodies, but also to initiate their absolute state with respect to a given inertial frame \((XYZ)\). In particular, the dynamics of the system are investigated when a predefined angular momentum is given about the origin \( O \) of the aforementioned inertial frame. In this case, all bodies rotates about \( O \) with a constant angular velocity \( \Omega \). System (1) can be equivalently written in a rotating frame \((xyz)\) that rotates with angular velocity \( \Omega \) with respect to the inertial frame

\[
m_i \ddot{r}_i = G \sum_{j=1, j \neq i}^{N} \frac{m_i m_j}{\|r_{ij}\|^3} r_{ij} - m_i \Omega \times (\Omega \times r_i) - 2m_i \Omega \times \dot{r}_i \quad \forall i = 1 : N
\]

(2)

where \( r_i \) is the position vector of body \( i \) in the rotating frame and \( r_{ij} = r_j - r_i \). The effect of the rotation is included by adding Coriolis and centrifugal terms to the equation of motion of the \( i \)-th body. The addition of a predefined angular momentum to the system is used here to simulate realistic asteroidal aggregation process, e.g. re-accumulation dynamics of asteroidal fragments after a collision event. In this case, all fragments initially belong to a unique asteroid that spins with a certain angular velocity about its principal inertia axes. After a collision with an external body occurs, fragments are created and scattered away from the main body, but they keep part of the angular momentum they had before the collision, when they were part of a unique body. The results obtained in Section 4 can be applied to the subsequent re-accumulation phase, as well as to other asteroid aggregation scenarios.

The code is classified as a numerical integrator, since it implements all \( N \) to \( N \) gravitational interactions between bodies. More in detail, the code implements equations (2). From the numerical point of view, system (2) is an initial value problem and its solution strongly depends on the choice of the initial condition set, namely the initial relative state of each body and angular velocity of the rotating frame. An important feature of gravitational interaction, to be considered from the numerical point of view, is that it is characterized by slow dynamics: for the case of \( N \) bodies, the shortest characteristic time can be estimated as follows [11]

\[
T \sim \frac{1}{\sqrt{G\rho}}
\]

(3)

where \( \rho \) is the material density of the bodies. A constraint for the integrator can be derived from (3): since the dynamics are slow, to catch the dynamical behavior of the system, there is no need of having small time steps of integration. In particular the integrator time step shall satisfy

\[
t_{\text{step}} < \frac{T}{2} = \frac{1}{2\sqrt{G\rho}}
\]

(4)

For typical values of asteroid material density ranging from 1000 to 4000 kg/m\(^3\) [7], the maximum time step results in the order of \( 10^3 \) s.

2.2 Contact dynamics

With respect to collision and contact dynamics, the \( N \) bodies are treated as rigid bodies of complex shape. The assumptions and models used to handle contact dynamics are shortly described here. The interested reader can refer to [20, 21, 22, 23] for further details and discussion.
Since it is not practical to search for contact points for all possible pair of bodies, collision detection is implemented into two steps. A preliminary stage, called broad-phase collision detection, identifies pairs whose bounding boxes are near enough; far pairs are discarded. In the second step, called narrow-phase, contacts are detected for the pre-selected pairs of bodies. Bodies can collide and re-bounce in collision types ranging from fully elastic to complete inelastic, depending on the selected restitution coefficient. In the present case, inelastic collision has been considered. Interactions between rigid bodies imply the presence of some constraints: two rigid bodies shall not penetrate each other and, if they are in contact, friction shall act at the interface. Generally speaking, at time $t$, the state of the system can be described by generalized coordinates $q(t)$ and generalized velocities $v(t)$. To mathematically implement non-penetration constraint, the function $\Phi(q)$ is defined such as

$$
\Phi(q) = \begin{cases} 
> 0 & \text{if bodies are separated} \\
= 0 & \text{if bodies are in contact} \\
< 0 & \text{if bodies are interpenetrating} 
\end{cases}
$$

For such a function, the non-penetration constraint becomes $\Phi(q) \geq 0$. Frictional constraints are described by conic constraints. The Coulomb friction model is used. When two bodies are in contact ($\Phi(q) = 0$), normal and tangential forces act at the contact point:

$$
F_N = \hat{\gamma}_n n \\
F_T = \hat{\gamma}_u t_1 + \hat{\gamma}_v t_2
$$

where $\hat{\gamma}_n$, $\hat{\gamma}_u$, $\hat{\gamma}_v$ respectively are the components along the normal ($n$) and tangential ($t_1$, $t_2$) direction. The force acting at the contact due to friction can be written as

$$
F_f = \hat{\gamma}_n n + \hat{\gamma}_u t_1 + \hat{\gamma}_v t_2 \in C
$$

where $C$ is a three-dimensional cone of slope $\arctan(\mu)$, with $\mu$ representing the Coulomb friction coefficient. While condition $\hat{\gamma}_n \geq 0$ must hold to correctly represent the physical phenomenon along the normal direction, the following constraint shall be satisfied to ensure the right representation of the physical phenomenon along the tangential direction

$$
(\hat{\gamma}_u, \hat{\gamma}_v) = \arg\min \sqrt{\hat{\gamma}_u^2 + \hat{\gamma}_v^2} \leq \mu \hat{\gamma}_n \hat{\gamma}_u t_1 + \hat{\gamma}_v t_2 \mathbf{v}_T
$$

where $\mathbf{v}_T$ represents the relative velocity at contact. Condition (9) guarantees that the reaction force is dissipative, since the tangential force is opposite to the relative tangential velocity at contact.

### 3 NUMERICAL SIMULATIONS

This Section presents the simulations performed to characterize the process of asteroidal aggregation from a cloud of boulders, highlighting the degrees of freedom of the problem and the assumptions made while selecting the case of study.

#### 3.1 Simulation parameters

Several parameters have to be set to initiate the simulation of the asteroid aggregation process. In particular, to simulate realistic scenarios it is important to carefully select the physical properties of the $N$ bodies and to consistently initiate their dynamics.

Initial conditions play an extremely important role when dealing with $N$-body dynamics. The set of initial conditions includes the initial state of all $N$ bodies. As described in Section 2.1, all particles are expressed in a rotating frame that rotates with a given angular velocity $\Omega$ with respect to an inertial reference. In addition, the relative state of each body, namely their position and
orientation, as well as their linear and angular velocity, are initialized with respect to the rotating reference frame. Initial conditions are given at time zero, or $t_0$ in the following.

From the numerical point of view, the choice of the number of bodies is crucial, since it has great impact on the computational effort. For an integrator, the cost of evaluating gravity is $N^2$. For this reason, typical capabilities of numerical integrators are often limited to few hundreds of bodies. The computational cost can be reduced by clustering the gravitational effect of bodies through domain decomposition. In this case, the cost can be reduced to $N \log N$ (N-body simulators [11]). In the proposed implementation, bodies of complex shape are considered: each body is built as a convex hull enveloping a set of randomly generated points.

### 3.2 Case of study

The case of study is presented in this Section. The parameters have been selected to simulate the aggregation process of common asteroids of small/medium size, with a characteristic size of hundreds of meters. The physical properties of the asteroids are chosen among typical values of objects belonging either to the main asteroid belt or to the Near Earth Asteroids (NEA) population.

Table 1 summarizes all the simulation parameters used to obtain the results presented in Section 4. In the simulations, 200 bodies are randomly generated in a three-dimensional cube whose side is 5 km long. Bodies are medium size boulders of 130 m characteristic size on average; the size of the smallest ones is 40 m. As mentioned earlier, this is likely the case of re-accumulation after fragmentation due to a collision event. The aggregate is then a “rubble pile” of loosely aggregated boulders. The material density has been set to 3000 kg/m$^3$, which is typical for metallic-based asteroids [7]. As discussed in Section 2.1, gravity interactions are slow; the integrator time step can be of the order of $10^3$ s. However, a time step of 10 s is chosen, to correctly integrate collision dynamics. In fact, they are faster and need shorter integration time steps. With these figures, aggregation processes reach a stable configuration after a transient of few tens of hours, which correspond to computational times in the order of a few minutes.

#### Table 1. Simulation parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>number of bodies</td>
<td>$N$</td>
<td>200</td>
</tr>
<tr>
<td>characteristic size of bodies [min-average]</td>
<td>$L^*$</td>
<td>40-130 m</td>
</tr>
<tr>
<td>material density</td>
<td>$\rho$</td>
<td>3000 kg/m$^3$</td>
</tr>
<tr>
<td>initial position cube side length</td>
<td>$L_0$</td>
<td>5 km</td>
</tr>
<tr>
<td>integration time step</td>
<td>$t_{\text{step}}$</td>
<td>10 s</td>
</tr>
</tbody>
</table>

Simulations are performed for different sets of initial conditions. In particular, the norms of the linear and angular velocity of each body range from 0 (no relative motion between bodies), to the maximum values specified in Table 2. Similarly, different cases have been explored concerning the norm of the angular velocity of the rotating frame, which ranges from 0 to the maximum value in Table 2. Maximum values specified in Table 2 are identified as thresholds above which there is no aggregation of the particles, which are scattered apart from each other by too high relative velocities or centrifugal force.

#### Table 2. Range of values for initial conditions set.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>Max value</th>
</tr>
</thead>
<tbody>
<tr>
<td>linear relative velocity of bodies</td>
<td>$v_0$</td>
<td>0.15 m/s</td>
</tr>
<tr>
<td>angular relative velocity of bodies</td>
<td>$\omega_0$</td>
<td>$10^{-2}$ rad/s</td>
</tr>
<tr>
<td>angular velocity of rotating frame</td>
<td>$\Omega$</td>
<td>$5 \cdot 10^{-5}$ rad/s</td>
</tr>
</tbody>
</table>
Table 2 defines boundaries on the magnitude of the velocities. Without any loss of generality, the inertial frame \((XYZ)\) is oriented such that \(\Omega\) is directed towards the positive \(Z\) axis, with \(\Omega = \Omega \hat{Z} = \Omega \hat{z}\),

\[
\Omega = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}
\]  

(10)

Concerning the linear and angular velocity of each body, their direction is randomly generated.

The simulation campaign is performed by imposing to the system several initial condition sets, in order to cover different aggregation scenarios and to explore the combined effects between the initial conditions. Table 3 summarizes all simulation sets considered. IDs are specified for each simulation set, to be referred during the analysis of results (Section 4).

<table>
<thead>
<tr>
<th>Simulation set</th>
<th>ID</th>
<th>(v_0) [m/s]</th>
<th>(\omega_0) [rad/s]</th>
<th>(\Omega) [rad/s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 - no initial motion</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1 - single parameter</td>
<td>1.1</td>
<td>0 − 0.15</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1.2</td>
<td>0</td>
<td>0 − 10(^{-2})</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1.3</td>
<td>0</td>
<td>0</td>
<td>0 − 5 · 10(^{-5})</td>
</tr>
<tr>
<td>2 - double parameter</td>
<td>2.1</td>
<td>0 − 0.15</td>
<td>0 − 10(^{-2})</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>2.2</td>
<td>0 − 0.15</td>
<td>0</td>
<td>0 − 5 · 10(^{-5})</td>
</tr>
<tr>
<td></td>
<td>2.3</td>
<td>0</td>
<td>0 − 10(^{-2})</td>
<td>0 − 5 · 10(^{-5})</td>
</tr>
<tr>
<td>3 - all parameters</td>
<td>3</td>
<td>0 − 0.15</td>
<td>0 − 10(^{-2})</td>
<td>0 − 5 · 10(^{-5})</td>
</tr>
</tbody>
</table>

The case with no initial motion between bodies and no rotation imposed is marked ID ‘0’. The first simulation set (ID ‘1’) includes the analysis of the effect of one initial condition at a time, with the remaining set to zero. The second set (ID ‘2’) analyzes the effect of two parameters at a time. Finally, the third set of simulations (ID ‘3’) investigates the case of simultaneously fully perturbed initial condition space.

4 RESULTS

The results of the aggregation simulations discussed in Section 3 are shown here. The first part of the Section discusses the dynamical evolution of the system, up to the formation (or non-formation) of a stable aggregate (Section 4.1). The aggregation process is studied by looking at the time profile of the total angular momentum of the \(N\)-body system, and by monitoring the evolution of the relative position of all bodies. The last part of the Section focuses on what happens after the transient. In case the system has converged to a stable asteroidal aggregate, the resulting physical and geometrical properties are studied and compared to the properties of known asteroids. Shape and inertia properties of the aggregate and their dependence on initial conditions are investigated. Results are shown in Section 4.2 for all simulation sets. Significant quantities extracted from the simulation scenarios include the bulk density (mean density, including void fraction) of the aggregate, its inertial elongation (ratio between maximum and minimum principal inertia moments) and rotation state.

4.1 Aggregation dynamics

An example of aggregation sequence is shown in Figure 1 for case ID ‘0’. The \(N\) bodies are driven solely by their mutual attraction, with no effects due to rotation of the reference frame. When boulders start to interact, few small bodies are scattered away because of collisions, but a stable aggregate eventually forms, collecting 193 bodies out of 200.
Figures 2 and 3 show the time profile of the total angular momentum of the system, with respect to the origin of the inertial frame

\[ \mathbf{H} = \sum_{i=1}^{N} \mathbf{h}_i \]  

(11)

The norm and the three Cartesian components are shown as function of simulation time. Case ID ‘0’ is shown in Figure 2(a). At \( t_0 \), the system is at rest; after a few hours, the first collisions between bodies take place. Most of the collisions occur between 5 and 10 hours after \( t_0 \). After the transient, the system turns into a single aggregate. The system is initially at rest; after the aggregation process, it gains a very small rotational velocity: the aggregate rotates with a period of about \( 10^3 \) hours. A different case, extracted from set ID 1.3, is shown in Figure 2(b). In this case, the system is initially provided with a common angular velocity (\( \Omega = 1.5 \cdot 10^{-5} \) rad/s). The rotational energy of the system is partially dissipated by collision events (between 5 and 10 hours after \( t_0 \)), but in the end the final aggregate keeps most of it. The final cluster rotates with a period of a few hours (fast rotator).

![Figure 2](image1.png)

**Figure 2.** Total angular momentum of the \( N \)-body system: (a) case ID ‘0’ (b) case extracted from set ID ‘1.3’ (\( \Omega = 1.5 \cdot 10^{-5} \)).

Figure 3(a) shows a simulation case extracted from set ID 1.2. The bodies are provided with a
given angular velocity about their body axes ($\omega_0 = 4 \cdot 10^{-3}$ rad/s). The spinning state of each single body is transmitted between boulders through collisions up to the formation of a single aggregate with a nonzero angular velocity. In this specific case, the final aggregate is left with a rotation period of some tens of hours (slow rotator). The effect of combined initial conditions in $\omega_0$ and $\Omega$ is shown in Figure 3(b). The effect due to collisions of dissipating the kinetic energy associated with rotation and to transfer angular momentum among particles is balanced in this case. The angular momentum of the system stabilizes at about its initial value after the collisions transient. The resulting aggregate is a fast-spinning asteroid.

![Figure 3](image)

**Figure 3.** Total angular momentum of the N-body system: (a) $\omega_0 = 4 \cdot 10^{-3}$ rad/s (from set ID ‘1.2’) (b) $\omega_0 = 7 \cdot 10^{-3}, \Omega_0 = 1.5 \cdot 10^{-5}$ rad/s (from set ID ‘2.3’).

Maximum breakup values (Table 2) identify the values of $v_0$, $\omega_0$ and $\Omega$ above which no aggregation occurs for the case of study specified in Table 1. The results show good agreement with data from the literature [7], scaled to this case of study. The formation of more than one aggregate has been observed when initial conditions are close to their maximum breakup values. In some cases, initial conditions combine to peculiar configurations: Figure 4 shows two examples of the formation of a binary asteroid system (two main aggregates).

![Figure 4](image)

**Figure 4.** Examples of binary asteroid formation, extracted from simulation set ID 2.3 and ID 2.2.

### 4.2 Properties of the final aggregate

Different kinds of aggregates have been obtained from the simulation campaign. This Section identifies the properties of the aggregates and compares them with known asteroids to verify the
ability of the numerical code of representing gravitational aggregation phenomena.

After its stabilization, the aggregate is considered as a single asteroid. The shape of the aggregate is found by enveloping it using an alpha shape algorithm [24]. Intuitively, the alpha shape algorithm finds the enveloping surface of the aggregate by rolling a sphere of radius $\alpha$ over the cluster of points. The value of $\alpha$ influences the final result, by constraining the path of the rolling sphere, with $\alpha = \infty$ being equivalent to the convex hull representation. Once the asteroid is identified, its characteristics are studied. Significant properties of asteroids are here briefly defined. The first quantity considered is the inertial elongation (or simply elongation in the following) $\lambda$, that is defined as the ratio between the maximum and the minimum principal inertia moments. It is always greater than or equal to one; the larger $\lambda$ is, the more elongated is the asteroid. Note that this property refers to the mass distribution of the asteroid, not to its geometry. An important quantity for the case of asteroids is the bulk density $\rho_b$, that refers to the mean density of the asteroid, including internal voids. Accordingly, the porosity $P$ is defined as

$$P = 1 - \frac{\rho_b}{\rho}$$

where $\rho$ is the material density (Table 1). In the following, $T_{agg}$ is used to indicate the period of rotation of the aggregate, $M_{agg}$ its total mass, and $N_{agg}$ the number of bodies (out of 200) in the final aggregate.

![Figure 5](image)

**Figure 5.** Characteristics of the final aggregate as function of normalized initial conditions: simulation results and best fitting curves.

Figure 5 shows how inertial elongation, rotation period and number of bodies in the aggregate are affected by initial conditions on $v_0$ (blue asterisks), $\omega_0$ (green stars) and $\Omega$ (red diamonds) for the case of study (Table 1). Exponential or polynomial fitting curves are displayed to separately show the trend of each contribution. The values of the initial conditions refer to a normalized range on the abscissa: each parameter ranges from 0 to 1, being 0 their minimum value ($v_0 = \omega_0 = \Omega = 0$)
and $I$ their maximum value \( (v_0 = v_{0\text{max}}, \omega_0 = \omega_{0\text{max}}, \Omega = \Omega_{\text{max}} \text{ from Table 2}) \). The upper plot shows that high elongations are obtained for high $\Omega$ or for low $v_0$, while no precise trend can be derived for $\omega_0$. On the other plots, all parameters are observed to share a common trend characterizing their effect on rotation period and number of bodies in the aggregate. As far as the former is concerned, small relative motion and rotation produce slowly rotating asteroids and vice versa: fast rotators are formed when the bodies are initialized with high angular and linear velocities. In particular, within the validity domain of this case of study, $\Omega$ is observed to produce the strongest effect and $\omega_0$ the weakest one. As for the last case, the number of bodies aggregating in the final asteroid decreases as velocities increase, with $\omega_0$ playing a dominant role.

Figure 6 shows simulation results that correlate the rotation period of the asteroid to its elongation and total mass. Results are shown on semi-logarithmic plots (the rotation period scale is logarithmic). A clear trend can be extracted from the plot on the right: as far as gravitational aggregates are concerned, smaller asteroids rotate faster than more massive ones. This is found in agreement with balancing between centrifugal force and gravitational attraction predicted by the theory [7].

![Figure 6](image.png)

Figure 6. Simulation results: rotation period as function of inertial elongation and total mass in the asteroid population.

Figures 7 and 8 show examples of asteroids obtained by enveloping all aggregating bodies after the dynamical transient. Minimum (Figure 7(a)) and maximum (Figure 7(b)) elongation cases are shown, in a population of asteroids ranging from quasi-spherical shapes ($\lambda_{\text{min}} \simeq 1.15$) to highly elongated ones ($\lambda_{\text{max}} \simeq 2.7$).

For what concerns the porosity (or equivalently the bulk density) of the aggregates, the results show that very little variability exists for the case of study. The porosity is between 34% and 40% for all asteroids, corresponding to $\rho_b \simeq 1900 \text{ kg/m}^3$. This is found to be in good agreement with theory and observations [7]. The only exception among the asteroid population is observed for small aggregates (Figure 8(b)), for which $P \simeq 14\%$, corresponding to $\rho_b \simeq 2500 \text{ kg/m}^3$. This result is also found in agreement with observations: smaller asteroids are more compact, with a lower fraction of interior voids [7].

5 CONCLUSIONS

Significant scenarios have been analyzed in the framework of realistic asteroid formation processes. Different sets of initial conditions have been investigated. The initial dynamical state of the $N$ bodies is found to play a fundamental role in the evolution of the cloud of particles. In detail, the state of the $N$ bodies has been initialized by either imposing no initial relative motion between them, or by imposing a given rotational and relative motion. The presented results show good agreement with theoretical predictions and observations, and suggest the ability of the numerical
code to predict natural aggregation phenomena.

The cost of evaluating gravity is $N^2$. It can be reduced to $N \log(N)$ by clustering the interactional effect between far clusters of particles. The current phase of the project entails the setup of the procedure. For this reason, no optimization has been performed yet. The partitioning of the domain using octrees, and the GPU-based parallelization of gravitational and contact forces computation will be pursued in a subsequent phase. This promises to grant the capability of handling a higher number of bodies, extending the possibility to simulate different dynamical scenarios such as impact and disruption ones.

Future applications will include the simulation of orbital dynamics about gravitational aggregates. The outcome of the aggregation process will serve as a high-fidelity model of the asteroid’s mass distribution. Space mission scenarios will be implemented to compute and simulate trajectories of spacecraft under the complex gravity field of such highly irregular bodies.

REFERENCES


Planning and Control of Autonomous Underwater Vehicle’s Trajectory for its Recovery from a Mobile Submarine Platform

Pedro Martínez*, José L. Aguirre*, Mariano Saura*, Pablo Segado*, Alberto Luaces#

* Department of Mechanical Engineering
Universidad Politécnica de Cartagena
Doctor Fleming, s/n, 30202 Cartagena, Spain
[joseluis.aguirre,msaura.sanchez]@upct.es
[pmi,psc1]@alu.upct.es

# Mechanical Engineering Laboratory
University of La Coruña
Mendizábal, s/n, 15403 Ferrol, Spain
aluaces@udc.es

ABSTRACT

Autonomous underwater vehicles (AUV) have seen a very strong growth over the last decade and, due to the high performance of submarine electronics and navigation systems, AUV applications grow constantly in many different sectors. Nevertheless, risks associated with their recovery exist, and there is a strong interest in the development of effective methods and algorithms to assist in this complex manoeuvre. In this paper, two algorithms that plan and control the trajectory for the underwater recovery of an AUV from a mobile platform have been developed and implemented: the first one plans the AUV trajectory and velocity to reach the mobile target in specific conditions and, the second one, controls the AUV trajectory to follow the path, while checking errors in position, orientation and velocity. The main characteristic of the trajectory is that it has been defined in relative coordinates with respect to the mobile platform; this offers a number of advantages with respect to global trajectories, as obstacle avoidance while the platform moves. The algorithm effectiveness, applicable to any AUV, has been tested by dynamic simulations of the REMUS100 AUV, considering variations in the following variables: initial position and orientation of the AUV, velocity and trajectory of the mobile platform, and refresh rate of navigation system measurements. From the results, it can be inferred that the developed algorithms are able to plan a trajectory in a wide range of initial conditions and to control the vehicle during the whole trajectory with errors under 0.4 meters in position and 10 degrees in orientation.

Keywords: Trajectory planning and control, Autonomous underwater vehicle, Recovery submarine platform, Dynamic simulation

1 INTRODUCTION

In the Naval industry, there are two fundamental trends that are developing rapidly: to provide a new generation of submarines with an increased payload capacity, and to increase the safety of personnel by using unmanned vehicles. To this end, the US Master Plan [1] has identified up to nine functions in which an autonomous underwater vehicle (AUV) is superior to a conventional submarine. Among these nine functions, the most important are: Intelligence, Surveillance and Reconnaissance (ISR), Mine, Oceanography and Networks Communication / Navigation.

Advances in underwater electronics, onboard energy management, as well as hardware and software for acquisition, processing and storage of large amounts of information, endow these vehicles with such a high performance that they have caught the attention of different kind of organisations: Army, research centers and private companies, to name a few. Part of these achievements are due to the ability of these vehicles to perform long-distance missions autonomously, as they count on the latest developments in sensors: Inertial Measurement Unit (IMU) to estimate spatial position and orientation, Doppler velocity log (DVL) for the measure of depth, temperature sensor, etc; as well as the most advanced navigation systems: Ultra Short Base Line (USBL) to locate a transmitter unit or Obstacle Avoidance Sonar (OAS). In addition, AUVs have control algorithms that allow them to follow a predefined path loaded on their system.
However, the recovery of AUVs from a mobile platform, which is a very interesting problem nowadays, is not solved despite the mentioned technological advances. This is mainly due to the fact that the waypoints that define the trajectory of the AUV are specified with respect to a non-inertial coordinate system (not useful to define the trajectory to reach a mobile platform), to the narrow ranges in the accuracy of the sensors required to avoid obstacles and prohibited navigation zones, and to the limited bandwidth in underwater communications which significantly reduces the possibilities of an accurate guidance and control of the AUV towards a recovery point.

This paper presents a solution to this problem, based on the planning and control of the trajectories defined in terms of relative positioning with respect to the mobile platform. In this sense, the recovery trajectory can be loaded in the AUV system so that, independently from the movement of the platform, the AUV will reach the recovery point while avoiding the predefined obstacles and forbidden navigation areas. The main objective of this work is to define and describe the developed algorithms for trajectory planning and control on an AUV that must be recovered in a moving platform, and to study the stability of the algorithms under variations in the recovery conditions. To do this, in section 2 the dynamic model of the AUV considered in the simulations is presented, and the algorithms for trajectory planning and control are described, as well as the simulations carried out to evaluate the effectiveness of the implemented algorithms. Finally, section 3 shows the simulation results, and the last two sections include the main conclusions of this work and future developments, respectively.

2 MATERIAL AND METHODS

This section describes the dynamic model of the AUV used in the study, the algorithms implemented for trajectory planning and control, and the analysis performed to evaluate the effectiveness of the proposed methods.

2.1 Dynamic model of the AUV

The effectiveness of the planning and control algorithms proposed in this paper is evaluated by simulating the dynamics of the AUV in an underwater environment. Without loss of generality, the methodology followed in this section (thoroughly described in [2]) considers the AUV REMUS100 whose dynamic equations can be found in [3].

2.2 Trajectory planning

The definition of the trajectory that allows the AUV to reach the mobile target is divided into two phases: check the starting position, and plan the trajectory.

**Check the starting position.** The vehicle, after its mission has been finished, navigates towards the starting zone, whose position and size are perfectly defined with respect to the recovery point. In the simulations carried out in this work, a prismatic zone has been defined: at 100 m behind this reference point, it extends up to 300 m (its length is 200 m), it has a width of 240 m (±120 m) and a height of 80 m, (±40 m). The AUV remains in this area, waiting for a ping call from the platform to indicate the start of the recovery manoeuvre. An array of hydrophones (commercial communication technology USBL -UltraShort Base Line-) in the AUV evaluates its position and orientation relative to the target and whether it is within the recovery starting area. If the AUV is not within that zone, it will navigate towards that zone.

**Trajectory planning.** Considering the current position and direction of the AUV with respect to the platform, the speed at which the AUV must navigate among the different sections of the path, and the speed and trajectory of the moving platform, the AUV runs a path planning algorithm that will allow it to reach the destination, while avoiding certain navigation areas defined previously. This path is defined in three steps (Fig.1):

**Stage 1 (P₀P₁):** layout with clothoids and circular curves (based on road layout [4]) allows the
Figure 1: Horizontal view of the three legs of the planned trajectory and the real trajectory followed by the AUV when the target was moving at the speed of 1 knot.

AUV to get oriented in the same direction as the target, keeping its depth during all this part of the trajectory.

Stage 2 ($P_1P_2$): using cubic spline curves, the AUV is directed to a predefined position next to the target and at its same depth (In Fig. 1 only the horizontal view is shown). In this work, this position has been defined at (-48,-41, 0) m relative to the target.

Stage 3 ($P_2P_3$): defined, in advance, in the horizontal plane through clothoids and circular arcs and with respect to a coordinate system fixed to the target.

The path planning algorithm takes into consideration that the maximum curvature does not exceed that permitted for the AUV. Through dynamic simulation (turning circle manoeuvre) it has been obtained a radius of curvature of 6 m for the REMUS100. In order the AUV to follow the planned trajectory, a number of interpolated waypoints are defined together with the speed that the AUV must have, at these points, relative to the target.

2.3 Trajectory control system

The implemented control system acts on the three governing elements of the AUV: pitch rudder, heading rudder and propulsion system, depending on the errors in position, orientation and velocity calculated by comparing the values obtained by its navigation systems (IMU + USBL) with the planned values on its path (Fig.2). This control system comprises three basic elements: calculation errors, measurement systems (position, orientation and speed, IMU + USBL), and PID controllers.

I. Navigation systems: IMU + USBL. At the times indicated by the refresh rate of the USBL, the control algorithm evaluates the position and orientation of the AUV relative to the platform and adds a random error whose order of magnitude coincides to that of commercial devices (0.1° and 0.01m [5]). Out of these refresh intervals, the AUV only uses the IMU to estimate its position and orientation.

The IMU operates constantly. The linear acceleration and angular velocities with respect to local axes, affected by a random error, are integrated to obtain continuous estimates of the position and orientation of the AUV. The 1750-IMU from KVH Industries, Inc. (with errors: ±2mg and ±2°/hr) has been considered for the launched simulations.

II. Error calculation. This module determines the errors in position, velocity and orientation of the AUV, from the differences between their values during the planned trajectory and the values estimated by the navigation system of the vehicle (IMU and USBL).
Measure systems of position, bearing and speed
IMU + USBL

Error
PID controllers

Movement of UUV

Target position and movement

Figure 2: Diagram that shows, together with the AUV dynamic model, the different elements in the control system: calculation errors, measurement systems (IMU + USBL), and PID controllers.

**Position error:** The vertical plane is considered as the one that forms the local vertical axis $z$ and the line tangent to the trajectory at each point; the horizontal plane is the one that contains this tangent line and is perpendicular to the vertical one. The error in the vertical plane $|\vec{e}_z(t)|$ is calculated (Eq.2) as the difference in depth between the trajectory point $(x_i, y_i, z_i)$ and its current estimated position $(x_{auv}, y_{auv}, z_{auv})$. In the horizontal plane (Fig.3.a) the position error $|\vec{e}_h(t)|$ is calculated as the minimum distance between the estimated position of the bow $(x_{auv}, y_{auv})$, and the trajectory point $(x_i, y_i)$, as shown in Eq. 1.

$$|\vec{e}_h(t)| = [x_{auv} - x_i, y_{auv} - y_i]$$  

$$|\vec{e}_z(t)| = [0, z_{auv} - z_i]$$

**Orientation error:** It is defined as the angular difference between two vectors: the relative velocity of the AUV with respect to the target $\vec{V}_{hr}$ and the direction of the vector tangent to the trajectory at the point of minimum distance $(x_i, y_i)$, which projected to the horizontal and vertical planes allows us to calculate $\xi_h(t)$ and $\xi_z(t)$, respectively. Figure 3 shows the vectors involved

Figure 3: In the horizontal plane it is shown: (a) the variables involved in the position error calculation and, (b) the variables involved in the velocity error calculation.

To strengthen the control system, the trajectory points that the AUV has left behind are no longer considered; that is, if the distance from the bow of the vehicle to the trajectory point $n$ is larger than the distance to points $n + 1$ and $n + 2$, then point $n$ is excluded from the trajectory.
in the definition of the horizontal orientation error, and Eq.3 shows the analytic expression to calculate it.

\[ \xi_h(t) = \arg(\bar{V}_r^h) - \tilde{\alpha}_i^h \]  

With:

\[ \bar{V}_r^h = \bar{V}_{auv}^h - \bar{V}_i^h \]

where: \( \bar{V}_{auv}^h \) is the velocity of the AUV, \( \bar{V}_i^h \) is the velocity of the \( i^{th} \) point of the trajectory and \( \tilde{\alpha}_i^h \) is the orientation of vector \( tg_i \); all of them projected to the horizontal plane, which is fixed to the movable target (in all the simulations, this plane coincides with the horizontal plane of the non inertial reference system).

**Velocity error:** Together with the position and orientation errors, it has been defined an error in the velocity of the AUV at which it might reach the target. This error \( e_v \) is calculated (Eq.5) as the difference between the velocity \( \bar{V}_{tri} \) that the AUV might have at each point of the trajectory \((x_i, y_i)\), relative to the target, and the projection \( \text{Proy} \bar{V}_r \) of the relative velocity of the vehicle to a vector which is tangent to the trajectory curve at this point. The velocity error is calculated in both the horizontal and the vertical planes of the non inertial coordinate system. The vectors and parameters involved in the horizontal plane are shown in Fig.3.b.

\[ \Delta = \bar{V}_{tri} - \text{Proy} \bar{V}_r ; \quad e_v = \text{sign}(\Delta) \cdot |\Delta| \]  

### 2.4 Trajectory control

Three PID controllers (proportional - integral - derivative) act on the orientation of the bearing and pitch rudders, as well as on the rotation speed of the propeller: yaw, pitch and speed of AUV, respectively.

The bearing (yaw) controller evaluates the orientation of the corresponding rudder \( \lambda_{ei} \), depending on the position \(|\bar{e}_h(t)|\) and the orientation \( \xi_h(t) \) errors in the horizontal plane, as defined in Eq. 6, where \( K_{pd} \) and \( K_{pa} \) are the proportionality coefficients, \( T_{dd} \) and \( T_{da} \) are derivative time constants, and \( T_{id} \) and \( T_{ia} \) are integral time constants, defined to calculate position errors (subindex \( d \)) and orientation errors (subindex \( a \)), respectively. Moreover, \( \alpha \) and \( \beta \) parameters define the distributed weight of both controllers.

\[ \lambda_{ei} = \alpha K_{pd} |\text{sign}(|\bar{e}_h(t)|)|\bar{e}_h(t)| + T_{dd} \frac{d\bar{e}_h(t)}{dt} + \frac{1}{T_{id}} \int \bar{e}_h(t) dt + \beta K_{pa} [\xi_h(t) + T_{da} \frac{d\xi_h(t)}{dt} + \frac{1}{T_{ia}} \int \xi_h(t) dt] \]  

On the other hand, the pitch controller evaluates the orientation \( \tilde{\lambda}_{zi} \) of the corresponding rudder depending on the position \(|\bar{e}_z(t)|\) and orientation \( \xi_z(t) \) errors in the vertical plane, as defined in Eq.7, where: \( K_{pd}^z \) and \( K_{pa}^z \); \( T_{dd}^z \) and \( T_{da}^z \); \( T_{id}^z \) and \( T_{ia}^z \) and \( \alpha^z \) and \( \beta^z \) have the same meaning than the ones stated for the horizontal plane.

\[ \tilde{\lambda}_{zi} = \alpha^z K_{pd}^z |\text{sign}(|\bar{e}_z(t)|)|\bar{e}_z(t)| + T_{dd}^z \frac{d\bar{e}_z(t)}{dt} + \frac{1}{T_{id}^z} \int \bar{e}_z(t) dt + \beta^z K_{pa}^z [\xi_z(t) + T_{da}^z \frac{d\xi_z(t)}{dt} + \frac{1}{T_{ia}^z} \int \xi_z(t) dt] \]
The velocity controller acts over the velocity \( n_{rpm} \) of the propulsion system (Eq.8), where \( K_p \), \( T_d \) and \( T_i \) have the meanings already mentioned.

\[
n_{rpm} = K_p [e_v(t) + T_d \frac{de_v(t)}{dt}] + \frac{1}{T_i} \int e_v(t) dt \tag{8}
\]

All the parameters in Eq.6 - 8 have been tuned by trial and error tests, trying to find a compromise between stability and responsiveness. For the bearing and pitch controllers, the tuning procedure is as follows: in the first place, with the orientation controller deactivated, the position parameters have been tuned by checking that the error along the whole trajectory was minimized. Then we proceeded in reverse order. Finally, the weight distribution of the position and orientation controllers has been determined from the error distribution along the whole trajectory, with special focus on the errors when the target point is reached.

As a final consideration, to achieve a more realistic simulation, kinematic constraints on the rudder movements have been defined: on the one hand, the gyration angle has been limited to ±20 degrees, and the angular velocity of the rudders has been limited to rates found in commercial equipment: 60° in 0.2 seconds for the angular velocity of the rudders, and a maximum of 100 rps or 200 rps, at each time step, when increasing or decreasing speed, respectively.

### 2.5 Simulation tests

To evaluate the effectiveness of the algorithms defined in this work, a series of dynamic simulations have been implemented under the Matlab programming environment. The algorithms must be able to plan a trajectory, taking into consideration the limitations of the selected vehicle, so that it can reach a specific target on the mobile platform with the lowest errors in position and orientation, avoiding specific areas in which the navigation of the AUV is forbidden and with the only assistance on its navigation systems: IMU and USBL. To check the effectiveness, different starting positions and orientations, as well as trajectories with different radius of curvature and speeds of the mobile platform, have been considered.

As forbidden navigation areas, the AUV must reach a target point in the interior of a rectangular prism cage with an open lateral access, and whose main dimensions are defined in Fig.4.

Limits for the maximum errors in position (±0.4 m) and orientation (±10°) at the target point have been defined. These limits are more restrictive than the values found in the literature for this underwater vehicle [6]. Both the position and orientation errors in the horizontal and the vertical plane will be studied under these conditions: USBL refresh rates (0.1 s and from 0.5 s to 3.0 s with increments of 0.5 s), different velocities (0, 1 and 2 knots) and radius of curvature (100 m, 500 m and ∞) of the mobile platform. In all the simulations, commercial errors for the instrumentation USBL and IMU have been already defined: ±0.1°, ±0.01 m and ±2 mg, ±2°/hr, respectively. In order to study the error dispersion, each test has been carried out up to five times.
3 SIMULATIONS RESULTS
This section shows and discusses the results obtained from the dynamic simulations that have been carried out.

3.1 Verification of the trajectory planning
Figure 5 shows the capability of the planning algorithm to define the trajectory that the AUV must follow to reach the mobile target. As the position of the AUV within the recovery zone is modified (Fig.5.a), the planning algorithm defines the first and second stages of the trajectory in order to achieve the starting point of the third stage. It can be seen that if this starting point is kept, the third stage remains always the same to ensure that the vehicle does not pass through prohibited navigation areas.

![Figure 5: Paths taken by the AUV by modifying different parameters of the trajectory.](image)

On the other hand, Fig. 5.b plots three trajectories planned by the algorithm by keeping fixed the starting position of the recovery manoeuvre, but varying the initial point of the third stage. As shown in the figure, if the point is defined such that the slope of the last stage is very low (solid line), the vehicle might collide with the aft part of the cage. However, even for significantly increased slopes, the AUV will enter in the cage, due to other parameters that define this final step. The entrance curve is determined from the minimum radius of curvature of the vehicle and the maximum centrifugal acceleration experienced by the AUV when it follows the trajectory at a constant speed.

3.2 Path-following errors
Figure 6 shows, as an example, the position and orientation errors of a REMUS100 simulation, both in vertical (pitch) and horizontal (yaw) planes versus the simulation time.

In the position error graph, small oscillations under ±0.2 m are shown along the entire path, somewhat higher in yaw at the first stages, because the AUV requires a period of adaptation to meet the target movement conditions.

The bearing error (Figure 6) has a more pronounced oscillation due to the constant adjustment of the aft rudders of the AUV to reduce position and orientation errors. The magnitude of these errors is small (±5°), indicating that the controller is fast and robust in sections with different radii of curvature.
3.3 Position and bearing errors at the target point

The calculation of the position and bearing errors at the target point allows to evaluate the effectiveness of the developed control algorithms. As indicated in section 2, these errors have been evaluated under different test conditions.

**Target moving at different velocities**

Under the conditions defined in the reference test (section 3.2), in these simulations the target moves in the longitudinal direction at constant speeds: 0, 1 and 2 knots.

According to Fig. 7.a, the vehicle is capable of reaching the target area no matter what the platform velocity is, with maximum errors in position around 0.1 m. There is a low dispersion in the results and, the lower the target velocity, the greater the sideways displacement of the error. This is
because, as the relative speed at which the AUV must move is set, the lower is the target velocity, the lower vehicle velocity corresponds and, therefore, the governing capabilities worsen, affecting specially at the stage that drives the AUV into the cage. Furthermore, the definition of a relative path softens the actual path of the AUV as the target velocity increases.

Regarding the bearing error in Fig. 7, the vehicle hits in the target area with a misalignment lower than $\pm 5^\circ$ for any of the tested velocities. These errors are very small and their dispersion is due to the natural oscillation of the system, as seen in Fig. 6. Nevertheless, it is possible to detect a trend towards a negative yaw misalignment trying to correct the yaw position errors.

**Target moving with different radius of curvature**

Maintaining the conditions of the reference test (section 3.2), trajectories of the moving platform are analyzed with the following radius of curvature: 100 m, 500 m and $\infty$ (straight line). In these tests, the AUV always approaches from the right side of the target which moves at 1 knot and turns to the left side.

![Figure 8: Final position and bearing errors of the REMUS 100 when the target describes three trajectories with different radii of curvature.](image)

Figure 8 shows how the vehicle hits in the target area with less misalignment than $\pm 5^\circ$ for the radii of curvature considered in the analysis. There is a small dispersion of the errors, which is more pronounced in yaw position and pitch orientation. Although the dispersion in yaw position can be explained because of the major complexity of the approaching manoeuvre as the target trajectory is more closed, the pitch error in orientation varies with independence of the test case. The variations are small, possibly due to instabilities of the AUV equilibrium in the vertical plane.

**3.4 Different refresh times of the USBL**

The AUV uses the IMU to update its position during the navigation, and the error suffers a drift in each iteration step until the USBL refreshes its position and orientation with high accuracy. In this test, maintaining the same reference conditions, the refresh time has been increased until the vehicle fails to hit the target or is unable to follow the path.

In Fig. 9 the vehicle does not hit the target area if the USBL refresh intervals are greater than 3 s. However, no more than $10^\circ$ misalignment are reached for any test conditions. The dispersion in position error is greater as the refresh rate of USBL signal increases. Additionally, the yaw error is more pronounced when the refresh time is also extended because the trajectory tracking in the last stage worsens, and this affects the AUV arrival.

Regarding the bearing error graph, it should be noted that, although the errors are within the limits.
in all the tests, only 0.1 s and 0.5 s values of refresh rate show a reduced dispersion. Above these values, it is not possible to recognise any variation pattern in the dispersion. A detailed study of the AUV behaviour along the whole trajectory for each refresh rate suggests that this is due to the control characteristics and the update instant of the USBL. When the signal is received on certain sections of the final curve, the system response oscillates slightly, thus producing failures in orientation. However, it has been proved that any attempt to adjust the controller in orientation causes a loss of accuracy in position.

4 CONCLUSIONS

This paper describes the algorithms for trajectory planning and control of an AUV which must pursue a mobile platform and reach a recovery point as accurately as possible in position, orientation and velocity. The trajectory planning algorithm is based on three specific stages which guide the AUV from a remote point, where the recovery manoeuvre starts, to the target point, where it ends; these stages are defined taking into account the vehicle manoeuvrability as well as the velocity and trajectory of the mobile platform. Since the planned trajectory for the AUV is defined relative to the platform, by simply varying the position of the starting point of the third stage, which can be done in advance, the AUV will avoid certain obstacles and prohibited navigation areas whose positions are known with respect to the same reference system.

The control algorithm acts over the depth and the bearing rudders, as well as over the propulsion system to control the pitch, yaw and the speed of the AUV, respectively. These algorithms use a weighted sum of errors, in position and orientation, to determine the angle of the corresponding rudder in the horizontal or the vertical planes. To evaluate these errors, the behaviour of two navigation systems, Inertial Measurement Unit and Ultra Short Base Line, both with commercial accuracies, has been implemented in the control loop.

To evaluate the efficiency of the developed algorithms, a dynamic model of the AUV has been implemented, and several simulations have been run, considering variations in the following parameters: starting point and orientation of the AUV, position and orientation of the AUV at the first point of the third stage, velocity of the AUV relative to the platform, velocity and trajectory of the mobile platform and refresh ratio of the USBL navigation system.

The trajectory planning algorithm defines a trajectory in three sections relative to the mobile platform; this offers the following advantages:

- Independent definition of the last section to ensure that the vehicle reaches the target point

Figure 9: Final position and bearing errors (REMUS100) with different USBL refresh rate.
while avoiding obstacles and prohibited navigation areas which are known in advance.

- A more accurate discretization in the last stage than in the other two, as greater AUV navigation restrictions exist, provides a better trajectory tracking and lower position and orientation errors at the target point.
- Precise tracking of the platform regardless of its trajectory and velocity, keeping small deviation in the error values at the recovery point.
- High flexibility in the trajectory definition as a wide range of initial AUV positions and orientations are allowed.

Regarding the adjustment and behavior of the control algorithm, the following conclusions can be drawn:

- Although the horizontal and vertical motion controllers are isolated, no error improvement has been observed when the vehicle moves only on the horizontal plane (first and third stage) with respect to when it moves relative to both of them (second stage).
- The controller parameters have been set by trial and error; although a systematic procedure has been followed, a complex relationship among the three parts (proportional, derivative or integral) makes difficult a perfect tuning of their corresponding parameters.
- It has not been possible to eliminate an oscillation of \( \pm 5^\circ \) in the orientation error along the whole trajectory; however, this error is below the allowable limits defined for a safe recovery.
- Small improvements are obtained in the position accuracy when the linear velocity of the target is increased, and the opposite effect is observed when the turning radius of the target path is reduced. However, despite these variations, in all the simulations the AUV hits the target with high accuracy.
- It has been found that changes in the update time of the USBL signal strongly affects the system accuracy: there is an upper limit of 3 s in the refresh rate. Above this value, the vehicle is unable to reach the target area.

5 FUTURE DEVELOPMENTS

During the analysis phase of the simulation results, some ideas have been raised that might improve the system response. With respect to trajectory generation, it has been found that the computational cost can be reduced if, instead of Euler spirals, B-spline curves were used, maintaining a similar behaviour from the point of view of the angular acceleration of the vehicle. In addition, trajectory planning might improve if changes in the velocity of the moving platform were taken into account. Finally, from the point of view of the controller, there are many ways in which it can be improved, as including Kalman filters into the control loop, adaptive control or fuzzy logic strategies, just to mention a few.

REFERENCES


Simulation of the flexible body moving in viscous fluid

Yaroslav A. Dynnikov¹, Galina Ya. Dynnikova², Sergey V. Guvernyuk¹,²

¹T-Services
Leninskii pr. 113/1, 117198 Moscow, Russia
yaroslav.dynnikov@gmail.com

²Research Institute of Mechanics
M. V. Lomonosov Moscow State University Michurinskii pr. 1,
119192 Moscow, Russia
[dyn, guv]@imec.msu.ru

ABSTRACT
Non-iterative method [1] is applied for solving the problem of strongly two-way coupled interaction between the flapping foil and incompressible viscous flow. For simulation of an incompressible flow the two-dimensional numerical method of viscous vortex domains (VVD) [2], [3] based on Navier-Stokes equations is used. The flexible foil is simulated with the model consisting of some sections which are connected via the elastic hinges. The comparison with the experiment is carried out for the rigid foil performing determined oscillations in the flow.

Keywords: Flow-structure interaction, vortex method, VVD, flapping foil, Navier-Stokes equations

1 INTRODUCTION
Studying of the movement of the elastic bodies in the flow is an actual problem of many engineering applications and bioengineering. The simulation of this movement is most difficult in the case of the strong two-way coupled fluid structure interaction. In this case, at each time step the hydrodynamic equations must be solved in accordance with the movement of the body, which changes the shape and position under the action of the hydrodynamic forces. Methods with splitting each time step in the two processes (body dynamics and hydrodynamics) is usually used with the following iterations [4]. Such an approach is not always effective, particularly in the cases where the rate of the relaxation for one of the processes is considerably less than of the other. Application of the gridless vortex method allows one to combine the equations determining the vorticity flux from the surface with the dynamic equations of the bodies motion in a general linear system of equations whose solution satisfies the boundary conditions on the surfaces of the velocity field without the splitting and iterations. In work [1] such approach was developed for the solving the fluid-structure interaction problem in the case of one rigid body (pendulum) in the fluid. Here the method is expanded for simulating the flexible body in the flow.

2 GOVERNING EQUATIONS
The flow of an incompressible fluid is described by the Navier-Stokes equations.

\[
\frac{dV}{dt} - \nu \nabla^2 V = -\frac{1}{\rho} \nabla p, \quad \nabla \cdot V = 0,
\]

where \( V, p, \rho, \nu \) are fluid velocity, pressure, density, and kinematical viscosity coefficient correspondingly.

In the case of the plane-parallel flows, it can be written as follows [5]

\[
\frac{dV_d}{dt} - V_d \times \Omega = -\frac{1}{\rho} \nabla p, \quad V_d = -\nu \frac{\nabla \Omega}{\Omega^2}, \quad \Omega = \nabla \times V,
\] (1)
Vector $V_d$ is called the diffusion rate.

The equation of the vorticity evolution, arising out of equations (1), has the form:

$$\frac{\partial \Omega}{\partial t} = -\nabla (u \Omega), \quad u = V + V_d. \quad (2)$$

Vector $J = V_d \Omega$ has the meaning the diffusion flux of the vorticity.

The no-slip condition is used on the surface, and no disturbance in the infinity.

The flexible bodies are modelled as consisting of solid sections connected by the elastic hinges (see Figure 1).

![Figure 1: The model of flexible foil.](image)

The force acting on each section is the sum of the hydrodynamic forces $F_H$ and the forces applied to the hinge axis $F_h$. It is supposed that there is no friction in the joint, and the restoring torque is proportional to the angle of deflection from the equilibrium state. The hydrodynamic forces $F_H$ and moments $M_H$ consist of the pressure and friction components $F_H = F_p + F_w$, $M_H = M_p + M_w$. The pressure force and moment acting on the part of the section surface between two points A and B can be expressed by applying integration by parts as follows:

$$F_p = \int_A^B p \mathbf{n} \, dl = \varepsilon_z \times \int_A^B \frac{d}{dl} r \, dl = \varepsilon_z \times (p_B r_B - p_A r_A) - \varepsilon_z \int_A^B \frac{\partial p}{\partial l} r \, dl$$

$$M_p = \int_A^B p (r - r_0) \times \mathbf{n} \, dl = \varepsilon_z \int_A^B \frac{\partial r}{\partial l} \, dl =$$

$$= \frac{\varepsilon_z}{2} \left( p_B (r_B - r_0)^2 - p_A (r_A - r_0)^2 \right) - \frac{\varepsilon_z}{2} \int_A^B \frac{\partial p}{\partial l} (r - r_0)^2 \, dl \quad (3)$$

From equation (1), the derivative $\partial p / \partial l$ is equal to

$$\frac{\partial p}{\partial l} = \frac{dr}{dl} \nabla p = -\rho \frac{dr}{dl} \frac{dV}{dt} + \rho \frac{dr}{dl} (V_d \times \Omega) =$$

$$= -\rho \frac{dr}{dl} \frac{dV}{dt} + \rho V_d \Omega \left( \varepsilon_z \times \frac{dr}{dl} \right) = -\rho \frac{dr}{dl} \frac{dV}{dt} + \rho J \mathbf{n} \quad (4)$$

In accordance with (4), the difference $p_B - p_A$ is equal to

$$p_B - p_A = \int_A^B \frac{\partial p}{\partial l} \, dl = -\rho \int_A^B \frac{dr}{dl} \frac{dV}{dt} \, dl + \rho \int_A^B J \mathbf{n} \, dl \quad (5)$$

Inserting (4) into (3) we obtain
The fluid velocity $V$ at the point $r$ on the body surface is equal to the velocity $V_s$ of the surface. It is the function of the velocity of the selected point $r_0$ of the body ($V_0$) and its angular velocity $\omega$.

\[
V_s = V_0 + \omega \times (r - r_0)
\]

The integrals, which contain the acceleration $dV/dt$, are the linear functions on $V_0$ and $\omega$ with the coefficients which can be obtained analytically.

3 NUMERICAL METHOD

For the flow simulation we use the VVD method [4], [5]. The equation (2) is solved in Lagrangian coordinates. The flow region with nonzero vorticity is represented as the set of the small regions (domains). The circulation of each domain is $\gamma_i = \Omega s_i$, where $s_i$ is the area of the domain. Such domain can be considered as the particle which moves at velocity $u = V + V_d$. During this motion, the intensity (circulation) of each particle remains constant. The method VVD is similar to the Diffusion Velocity method of Ogami and Akamatsu [6], but in VVD the other discrete formulas for calculation of the diffusion velocity are used. They are more correct especially near the surfaces.

The particles are generated at the nodes on the contour surface at each time step. The new vortex particles are determined from the no-slip boundary conditions which mean an equality of the fluid velocity $V$ at the surface with the surface velocity $V_s$. The velocity $V$ is determined by Biot-Savart formula which has the next discrete form

\[
V(r) = \sum_{i} v_i(r) \gamma_i + \sum_{k=1}^{N} v_k(r) \gamma_{k}^{\text{new}} + \\
\frac{1}{2\pi} \int_{C} (V_s \times n) \cdot \frac{(r - r_{e})}{|r - r_{e}|} dl - \frac{1}{2\pi} \int_{C} (V_n) \frac{(r - r_{e})}{|r - r_{e}|} dl,
\]

\[r_{e} \in C\]

$v_i(r)$ is the velocity induced by $i$-th particle with unit circulation at the point $r$, and $n$ is the unit normal to the contour $C$ directed into the body.

Boundary conditions are provided by the equations

\[n_j \bar{V}_j = n_j \bar{V}_j, \quad j = 1, 2, ..., N_j,\]  

(8)

where $\bar{V}_j$ is the average value of $V$ at $j$-th segment of the boundary contour. The no-slip conditions are satisfied because the attached vortices are absent at the surface. It is evident that equations (8) depend linearly on unknown values $\gamma_k^{\text{new}}$, $\bar{V}_{0,m}$ and $\bar{\omega}_m$, where $m$ is the number of the section. Using approximate expressions $\bar{V}_{0,m} = (V_{0,m} - V_{0,m}^{\text{old}})/\Delta t$, $\bar{\omega}_m = (\bar{\omega}_m^{\text{new}} - \bar{\omega}_m^{\text{old}})/\Delta t$, we
obtain linear equations for values $\gamma_{k}^{\text{new}}$, $\mathbf{v}_{0,m}^{\text{new}}$ and $\mathbf{w}_{m}^{\text{new}}$. The values $\gamma_{k}^{\text{new}}$ are connected with the flux of the vorticity $\mathbf{J}$ by the expression $\mathbf{J}_{n} = \gamma_{k}^{\text{new}}/(\Delta l_{i} \Delta t)$, where $\Delta l_{i}$ is the distance between the nodes. Hence the integrals in equations (5) (6) can be approximated by the sums

$$
\int_{A}^{B} \mathbf{J}_{n} \, d\mathbf{l} \approx \frac{1}{\Delta t} \sum_{j=1}^{j_{n}} \gamma_{j}^{\text{new}}, \quad \int_{A}^{B} (\mathbf{r} \times \mathbf{J}_{n}) \, d\mathbf{l} \approx \frac{1}{\Delta t} \sum_{j=1}^{j_{n}} \gamma_{j}^{\text{new}} \mathbf{r}_{j}, \quad \int_{A}^{B} (\mathbf{r} \times (\mathbf{r} \cdot \mathbf{J}_{n})) \, d\mathbf{l} \approx \frac{1}{\Delta t} \sum_{j=1}^{j_{n}} \gamma_{j}^{\text{new}} (\mathbf{r}_{j} \cdot \mathbf{r}_{i})^{2}
$$

So from (5) and (6) we have the expressions for the hydrodynamic forces and moments, which are linear on $\gamma_{k}^{\text{new}}$, $\mathbf{v}_{0,m}^{\text{new}}$ and $\mathbf{w}_{m}^{\text{new}}$. The friction forces don’t depend on the acceleration. The formulas for calculating $\mathbf{F}_{w}$ and $\mathbf{M}_{w}$ are presented in [3], [6]. The amount of the scalar dynamic equations for all sections is equal to $3N_{b}$, where $N_{b}$ is the number of sections. This equations include the unknown forces $\mathbf{F}_{h}$ acting in the axes i.e. there are additional $2(N_{b} - 1)$ unknown scalar values. To close the system, we use the constraint equations, which means a coincidence of the axes coordinates on the two adjacent sections. It should be noted that the unknown forces $\mathbf{F}_{h}$ are included in the equations in the combination with the pressure in the adjacent to the hinge points separating the sections. Solution of this system of equations gives us unknown quantities $\gamma_{k}^{\text{new}}$, $\mathbf{v}_{0,m}^{\text{new}}$ and $\mathbf{w}_{m}^{\text{new}}$, which simultaneously satisfy the boundary conditions and the dynamic equations of the bodies.

4 RESULTS AND DISCUSSION

The proposed method is applied to the problem of flapping foil in a flow. This task has been studied experimentally in papers [7], [8] and others. In paper [7] the rigid foil performing determined oscillations in a flow was investigated at different values of frequency and amplitude of angular oscillation. Different regimes of the flow behind the profile were observed such as usual Karman street, reverse Karman street which generates propulsive jet, and regimes with broken symmetry. In paper [8] the influence of the foil flexibility was studied. It was observed that flexibility inhibits the symmetry breaking, and increases propulsion force of the foil.

We have reproduced the experiments [7] numerically. The foil shape is the same as in paper [7]. It is shown in Figure 2.

![Figure 2. The profile shape.](image)

The flow velocity is $U$. Oscillation frequency is $f$. The problem is solved in non-dimensional parameters: the Reynolds number $Re = UD/\nu$, Strouhal number $St = fD/U$, non-dimensional amplitudes $A_{D} = Ad/D$. The obtained results are presented in the left side of Figure 3. In the right side the snapshots of the experiments are given. The numerical results are obtained at $St = 0.2$ and $Re = 250$ with varying amplitude. The black points represent vortices circulating clockwise and white points with a counter-clockwise circulation. In Figure 3a the amplitude $A_{D} = 0.36$. The wake is the usual Karman street. The vortices generated at the boundary layers on each side of the foil stay on the same side of the symmetry line of the wake (i.e. in Figure 3a the clockwise vortices are located above the counter-clockwise set). It has been shown in [2] that
when the amplitude or frequency increases then the Karman street transforms to reverse Karman street in which the rows of vortices change places. Before this transformation the transition regime exists when vortices are aligned in the symmetry line. The case of $A_D = 0.71$ in Figure 4b is close to such regime. In Figure 3c reverse Karman street is drawn (in the picture clockwise vortices are located below the counter-clockwise set). The average velocity near the symmetry line of the reverse Karman street exceeds the velocity of the incoming flow. This creates thrust. Further increase of the amplitude leads to the symmetry breaking, that is seen in Figure 3d at $A_D = 1.77$.

Comparison of the numerical results with the experimental data shows a good agreement.

![Figures 3](image)

Figure 3. Kinds of wakes. In the left side the results of calculations by VVD, in the right side experiments [7]: a) usual Karman street, $A_D = 0.36$; b) transition regime, $A_D = 0.71$; c) reverse vortex street $A_D = 1.07$; d) the symmetry breaking of the reverse vortex street $A_D = 1.77$.

4.1 The profiles with elastic hinges

Two types of the profiles are considered: consisting of one section at the elastic hinge, and consisting of three sections with three elastic hinges. Both profiles in the equilibrium state have the same shape as shown in Figure 2. The distance between the first and the second hinges is equal to the half of the distance $L$ between the first hinge and the profile end, and the distance between the second and third hinges is equal to $L/4$.

The holder performs determined oscillation $\alpha = \alpha_0 \sin(2\pi f t)$. All sections move under the action of flow and the hinge torques which are proportional to the angle differences $\Delta \alpha_1, \Delta \alpha_2, \Delta \alpha_3$, with coefficients $k_1, k_{II}, k_{III}$. For the both profiles $k_1 = 800 \rho D U^2$, and $k_{II} = k_1/4, k_{III} = k_1/16$.

In Figure 2 the foil shape and the vortex pictures at successive times are shown. The computations were carried out for the next values of parameters: $St = 0.5, \alpha_0 = 0.122$ (this corresponds to $A_D = 1$ for the fixed joint with the holder), $\rho = \rho_{body}/\rho_{fluid} = 1$, $Re = 250$, $t = \tau U/D$, where $\tau$ is physical time.

It is seen that the vortex wake is not symmetrical about the horizontal axis while the holder performs symmetrical oscillations. In paper [8] only symmetrical flows behind the flexible foil were observed, but the values of $St$ in their experiments were less than 0.45.
When the hinges are elastic then the oscillation amplitude of the foil tip $A_{\text{eff}}$ do not coincide with $A_D = L \sin \alpha_0$. In Figure 5 there are the dependences $A_{\text{eff}} / A_D$ on $St$ for two types of the foil at $\rho = 1$ and $\rho = 3$ shown. All other parameters are the same as above.

It is seen that there is the region of $St$ where the real amplitude exceeds the determined one. After reaching a maximum, the amplitude $A_{\text{eff}}$ decreases with increasing $St$ and becomes smaller than $A$. Such behavior of amplitude is typical for the resonance phenomenon. This
phenomenon was also reported in paper [9], where the numerical investigation of the thin flexible foil in an ideal flow was carried out. In the cases of the three-section foil the amplitude have additional local maximum which can be connected with resonance at the higher frequencies of second and third hinges.

In Figure 6 there are dependencies of the dimensionless drag force $F_x$ per unit of span referred to $\rho D U^2$ on Strouhal number shown. Here one can see that at the considered parameters the force is negative, i.e. thrust effect takes place. In all the cases there is a local maximum of thrust near $St = 0.5$. But in the case of one-section foil the thrust decreases after maximum with increasing $St$, while the thrust of the three-section foil behaves non-monotonically and increases again at $St > 1$.

![Figure 6. The dependency of the drag force on frequency](image)

5 CONCLUSIONS

The method presented is effective for solving flow-structure problems for rigid bodies with elastic connections. It is non-iterative and does not require the splitting of the time step to hydrodynamic and body dynamic parts when the problem of strongly two-way coupled interaction between the bodies and the fluid is solving. The method is applied for simulating the flapping flexible foil in incompressible viscous flow. Essential influence of the flexibility is demonstrated.

6 SUPPORT

The work was supported by the Ministry of Education and Science of the Russian Federation (Agreement No. 14.576.21.0079, project id RFMEFI57614X0079).
REFERENCES


Wave-Based Attitude Control of Spacecraft with Fuel Sloshing Dynamics

J. W. Thompson*, W. J. O’Connor#

School of Mechanical and Materials Engineering
University College Dublin
Belfield, Dublin 4, Ireland
* joseph.thompson@ucdconnect.ie
# william.oconnor@ucd.ie

ABSTRACT

Many mechanical systems are inherently flexible, making it difficult to achieve rapid, controlled motion. The control challenge is even greater when the system is not well modelled, has dynamics that change with time, or is under-actuated. A rocket with sloshing fluid propellant is an extreme case. Many control strategies struggle with such systems. However a wave-based control method has been shown to cope well with these challenges. The key idea is that the motion of the actuator can be separated into two notional components, one travelling from the actuator into the system, the other leaving the system through the actuator. Intuitively the actuator simultaneously launches mechanical waves into a system while it absorbs returning waves. When the launching and absorbing is finished vibrations have been damped and the desired reference motion is left behind. A mathematical model is developed for an upper stage accelerating rocket moving in a single plane. An equivalent mechanical model in the form of a pendulum is included to represent the fuel sloshing dynamics. In numerical simulations the controller successfully suppresses the sloshing motion. A major advantage of the strategy is that no measurement of the pendulum states (sloshing motion) is required. However it is found that when the effective sloshing mass becomes small relative to the rocket body, it takes longer to fully suppress sloshing motions. This is acceptable, however, because in this case, by definition, the sloshing does not cause a major problem for the rocket controller.

Keywords: Spacecraft Dynamics, Attitude Control, Flexible Systems, Sloshing, Mechanical Waves

1 INTRODUCTION

The failure of an early Jupiter ballistic missile in 1957 was caused by the interaction of the control system and the sloshing liquid fuel on board [1]. The problem was originally solved by aluminium drink cans, which could be fitted into the fuel tank and which floated on the surface of the fuel. Their friction both against the walls of the tank and against each other damped out the fuel oscillations. This experience motivated the eventual solution to the problem, which was the addition of baffles to the tank walls. However baffles can only provide a certain level of damping of the liquid motion and can only be optimized for one tank fill level, so their effectiveness is reduced as the propellant is depleted. They also add complexity and mass to the vehicle and so increase costs [2]. It therefore becomes necessary to design a control system for a rocket which actively takes into account the effects of propellant sloshing and tries to compensate for them. Many mechanical systems are inherently flexible, making it difficult to achieve rapid, controlled motion. The control challenge is even greater when the system is not well modelled, has dynamics that change with time, or is under-actuated. A rocket with sloshing fluid propellant on board is an extreme case of such a system. A wave-based control method has been shown to cope well with the challenges outlined above [3, 4]. The key idea is that the motion of the actuator can be separated into two notional components, one travelling from the actuator into the system, the other leaving the system through the actuator. Intuitively the actuator simultaneously launches mechanical waves into a system while it absorbs returning waves. When the launching and absorbing is finished, vibrations
have been damped and the desired reference motion is left behind. The method has been demonstrated to work well for 1-D and 2-D lumped flexible systems and in robotic and crane applications [5, 6]. The aim of this paper is to extend the application to the control of spacecraft with flexible structures and appendages (e.g. solar panels), and with on-board liquid propellant. This new area of application presents many new challenges. The spacecraft systems are often nonlinear, their associated flexibility is non-uniform, the sloshing dynamics are difficult or impossible to predict, and sensors and actuators can behave far from the ideal. In this paper the example of an accelerating upper stage rocket is examined. A wave-based controller is designed for the upper stage AVUM of the European launcher Vega and is tested by numerical simulation.

2 MATHEMATICAL MODEL
In this section a mathematical model is developed for an upper stage accelerating rocket moving in a single plane. The rocket is assumed to be in a microgravity environment and free from aerodynamic effects. The sloshing fuel mass is represented by a mechanical analog in the form of a simple pendulum attached to the main rocket body. The fuel mass is partitioned, according to the tank fill level, into a fixed point mass and moving pendulum mass [7].

2.1 Planar upper stage model
The model of the upper stage rocket with single pendulum is shown in figure 1.

The rocket body and pendulum are isolated and free body diagrams for each are shown in figure 3.

The equations of motion for the two bodies may be written as:

\[ F - R_x = ma_x \]  

(1)
\[ f - R_z = ma_z \]  
\[ I \ddot{\theta} = M + b(R_z) \]  
\[ R_x a \sin \phi + R_z a \cos \phi = 0 \]  
\[ m_f (a_z - b \dot{\theta}^2 + a \cos \phi (\dot{\theta} + \dot{\phi})^2 + a \sin \phi (\ddot{\theta} + \ddot{\phi})) = R_z \]  
\[ m_f (a_z - b \ddot{\theta} - a \sin \phi (\dot{\theta} + \dot{\phi})^2 + a \cos \phi (\ddot{\theta} + \ddot{\phi})) = R_z \]  

All symbols are described in table 1.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(m)</td>
<td>body mass</td>
</tr>
<tr>
<td>(I)</td>
<td>body moment of inertia</td>
</tr>
<tr>
<td>(m_f)</td>
<td>pendulum mass</td>
</tr>
<tr>
<td>(a)</td>
<td>length of pendulum</td>
</tr>
<tr>
<td>(b)</td>
<td>distance from COM to pivot point</td>
</tr>
<tr>
<td>(\theta)</td>
<td>body pitch angle</td>
</tr>
<tr>
<td>(\phi)</td>
<td>angle of the slosh pendulum w.r.t. body</td>
</tr>
<tr>
<td>(F)</td>
<td>axial force</td>
</tr>
<tr>
<td>(f)</td>
<td>lateral force</td>
</tr>
<tr>
<td>(M)</td>
<td>moment applied to body</td>
</tr>
<tr>
<td>(R_x)</td>
<td>internal constraint force at pivot point</td>
</tr>
<tr>
<td>(R_z)</td>
<td>internal constraint force at pivot point</td>
</tr>
<tr>
<td>(a_x)</td>
<td>axial body acceleration</td>
</tr>
<tr>
<td>(a_z)</td>
<td>lateral body acceleration</td>
</tr>
</tbody>
</table>

Substituting for \(R_x\) and \(R_z\) from equations 5 and 6 into equations 1-4 eliminates these internal constraint forces, and gives a minimal set of four equations for the four degree of freedom system.

\[ (m + m_f) a_x - m_f b \dot{\theta}^2 + m_f a \cos \phi (\dot{\theta} + \dot{\phi})^2 + m_f a \sin \phi (\ddot{\theta} + \ddot{\phi}) = F \]  
\[ (m + m_f) a_z - m_f b \dot{\theta} - m_f a \sin \phi (\dot{\theta} + \dot{\phi})^2 + m_f a \cos \phi (\ddot{\theta} + \ddot{\phi}) = f \]  
\[ (I + m_f b^2) \ddot{\theta} - m_f b a_x + m_f a b \sin \phi (\dot{\theta} + \dot{\phi})^2 - m_f a b \cos \phi (\dot{\theta} + \dot{\phi}) = M \]  
\[ (m_f a_x^2) (\ddot{\theta} + \ddot{\phi}) - m_f a b \sin \phi \ddot{\theta}^2 - m_f a b \cos \phi \ddot{\theta} + m_f a (a_z \cos \phi + a_x \sin \phi) = 0 \]

### 2.2 Choice of actuators

The model described is general in that the body is actuated by two forces, \(F\) and \(f\), and a moment, \(M\). In reality the rocket may have one or many actuators, but in any configuration these actuators may be resolved to these two forces, axial and lateral, and a moment applied to the rocket body. In some cases these inputs may not be independent of each other, but instead a function of some lesser number of inputs. For example, in the case of a rocket as shown in figure 3(a) with a single gimbaled engine the forces and moment are no longer independent and are given by:

\[ M = T (b + c) \sin \delta, \quad F = T \cos \delta, \quad f = T \sin \delta \]  

where \(T\) is the constant thrust developed by the rocket engine, \(c\) is the distance of the gimbal from the mass centre, and the single input is the engine gimbal angle \(\delta\). Similarly the rocket may be actuated by lateral thrusters as shown in figure 3(b). In this case:

\[ M = T d, \quad F = T, \quad f = T_l \]  

where again \(T\) is the constant thrust developed by the non-gimballing rocket engine, \(d\) is the axial distance from the thrusters to the mass centre, and the single input is the magnitude of the lateral thrust \(T_l\).
2.3 Linearized model

Solving equations 7 and 8 for $a_x$ and $a_z$ respectively gives:

$$a_x = \frac{F + m_f b \dot{\theta}^2 - m_f a \cos \phi (\dot{\theta} + \dot{\phi})^2 - m_f a \sin \phi (\dot{\theta} + \dot{\phi})}{m + m_f}$$  \hspace{1cm} (13)$$

$$a_z = \frac{f + m_f b \dot{\theta} + m_f a \sin \phi (\dot{\theta} + \dot{\phi})^2 - m_f a \cos \phi (\dot{\theta} + \dot{\phi})}{m + m_f}$$  \hspace{1cm} (14)$$

Substituting these expressions into equations 9 and 10 gives a simplified system of two equations describing the pitch and slosh dynamics:

$$\begin{bmatrix} I + mm* (b^2 - ab \cos \phi) \\ 0 \end{bmatrix} \ddot{\theta} - mm* ab \dot{\phi} \cos \phi + mm* ab (\dot{\theta} + \dot{\phi})^2 \sin \phi = M + m* b f$$  \hspace{1cm} (15)$$

$$m* (a^2 - ab \cos \phi) \dot{\theta} + mm* a^2 \ddot{\phi} + m* (aF - mab \dot{\theta}^2) \sin \phi = m* a f \cos \phi$$  \hspace{1cm} (16)$$

where:

$$m* = \frac{m_f}{m + m_f}$$  \hspace{1cm} (17)$$

After linearization about $[\theta, \phi, \dot{\theta}, \dot{\phi}] = 0$ equations 15 and 16 become:

$$\begin{bmatrix} I + mm* (b^2 - ab) \\ 0 \end{bmatrix} \ddot{\theta} - mm* ab \dot{\phi} = M + m* b f$$  \hspace{1cm} (18)$$

$$m* (a^2 - ab) \dot{\theta} + mm* a^2 \ddot{\phi} + m* aF \dot{\phi} = m* a f$$  \hspace{1cm} (19)$$

The state vector of the linearized system consists of the pitch and slosh angles and their derivatives:

$$\mathbf{x} = [\theta, \phi, \dot{\theta}, \dot{\phi}]^T$$  \hspace{1cm} (20)$$

It is assumed that the axial thrust $F$ is a constant, so that the input vector consists of the lateral force and moment $M$:

$$\mathbf{u} = [f, M]^T$$  \hspace{1cm} (21)$$

Now equations 18 and 19 may be rewritten in state space form:

$$\dot{\mathbf{x}} = \mathbf{A} \mathbf{x} + \mathbf{B} \mathbf{u}$$  \hspace{1cm} (22)$$

where:

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{F b m_f}{I (m + m_f)} & 0 & 0 \\ 0 & -\frac{F m_f (b^2 - ab)}{I a (m + m_f)} & -\frac{F}{a m} & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & \frac{1}{a m} \\ -\frac{1}{a m} \frac{(b^2 - ab)}{I a} \end{bmatrix}$$  \hspace{1cm} (23)$$
We find that the condition for stability of this linearized system is:

$$\frac{mmf(b^2 - ab)}{m + m_f} + I \geq 0 \quad (24)$$

Possible pendulum configurations are shown in figure 4. For $b < 0$ (pendulum pivot point behind the mass centre) (figure 4(a)) or $b > a$ (whole pendulum in front of mass centre) (figure 4(b)) this condition is always satisfied, but in the region $0 < b < a$ stability is achieved only if:

$$a - b \geq \frac{I(m + m_f)}{bmm_f} \quad (25)$$

Physically this means that the pendulum must straddle the point P shown in figure 4(c) which is a distance $-\frac{I(m + m_f)}{bmm_f}$ from the centre of mass along the x-axis.

3 WAVE-BASED MODEL

The first step in developing a wave model for the rocket system is to express the equations of motion in a form resembling a cascaded lumped flexible system. The rocket system described by equation 22 has two degrees of freedom and so it is required to transform this to appear like a 2-DOF lumped flexible system, i.e., two masses/inertias with an interconnecting spring. It is also required that there be a single control input which actuates just the first degree-of-freedom, i.e., a single launcher and absorber of waves. The system should have the following form:

$$\dot{z} = \hat{A}z + \hat{B}f_0 \quad (26)$$

where:

$$\hat{A} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{k_1}{m_1} & \frac{k_1}{m_1} & 0 & 0 \\ \frac{k_1}{m_2} & -\frac{k_1}{m_2} & 0 & 0 \end{bmatrix}, \quad \hat{B} = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{m_1} \\ 0 \end{bmatrix}, \quad z = \begin{bmatrix} x_1 \\ x_2 \\ x'_1 \\ x'_2 \end{bmatrix} \quad (27)$$

These equations describe the dynamics of a 2-DOF mass-spring system, where $x_1$ and $x_2$ are the displacements of the masses and $f_{act}$ is the actuating force on the first mass. Assume that the lateral force $f$ on the rocket body is zero and just the pure moment $M$ is available for control purposes. In reality this could be imagined as a gimballed rocket engine with lateral thrusters at mass centre to cancel the lateral forces from the engine. The input matrix $B$ from equation 23 then becomes a
column vector:

\[ B = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{(b-a)} \end{bmatrix} \]  

(28)

To transform the system to the required form the change of basis \( z = Mx \) is used, where:

\[
M = \begin{bmatrix}
1 & 0 & 0 \\
1 & \frac{a}{a-b} & 0 \\
0 & 0 & 1 \\
0 & 0 & 1 \frac{a}{a-b}
\end{bmatrix}, \quad \hat{A} = MAM^{-1}, \quad \hat{B} = MB
\]  

(29)

The states are now:

\[ x_1 = \theta, \quad x_2 = \theta + \left( \frac{a}{a-b} \right) \phi \]  

(30)

and the parameters of the system are given by:

\[ k_1 = \frac{F b m_f (a-b)}{a(m+m_f)} \quad m_1 = I, \quad m_2 = \frac{m m_f (b^2-ab)}{m+m_f} \]  

(31)

Figure 5 shows the equivalent mass-spring system with a notional mass \( m_0 \) and notional spring of stiffness \( k_0 \) appended to the system. The force in the first spring is considered to be the actuation force \( f_1 \). The system may now be considered as actuated by the displacement \( x_0 \) of notional mass \( m_0 \) such that:

\[ f_0 = k_0 (x_0 - x_1) \]  

(32)

The wave model assumes that the displacement of each mass \( x_i \) is can be seperated into leftward and rightward travelling components \( a_i \) and \( b_i \) respectively or \( A_i \) and \( B_i \) in the Laplace (com-plex frequency) domain [8]. The propagation of the rightward and leftward travelling waves is described by wave transfer functions \( G_i, H_i \) and \( F \) respectively such that:

\[ A_i = G_{i-1} A_{i-1}, \quad B_i = H_{i} B_{i+1}, \quad B_2 = F A_2 \]  

(33)

For controller design it is easier to work with transfer functions that deal with the actuating force \( F_0 \) rather than the notional displacement \( X_0 \). The spring force \( F_0 \) can also be seperated into rightwards and leftwards travelling components \( F_{0A} \) and \( F_{0B} \) respectively. Then the cross-over wave transfer functions \( P_0 \) and \( Q_0 \) relate displacements to forces by:

\[ A_1 = P_0 F_{0A}, \quad F_{0B} = Q_0 B_1 \]  

(34)

and these can be calculated from the ordinary wave transfer functions as:

\[ P_0 = \frac{G_0}{k_0 (1-G_0)}, \quad Q_0 = k_0 (H_0 - 1) \]  

(35)

4 CONTROL DESIGN

4.1 Wave-Based Controller

A WBC3 (force actuated) controller was designed to control the rocket attitude \( \theta \). The controller uses only the transfer functions \( G_0 \) and \( H_0 \). The second-order uniform system approximations [8] are used where:

\[ G_0 = \frac{\omega_G^2}{s^2 + \omega_C s + \omega_G^2}, \quad \omega_G = \sqrt{\frac{2k_0}{m_1}} \]  

(36)
The wave-based control scheme is shown in figure 6. $\theta_{ref}$ is the desired reference pitch angle. The control input is the input torque $M_{ref}$. Two variables are measured for feedback. These are the pitch angle $\theta$ and the actual achieved torque $M$. In this paper the actuator is assumed ideal except for the saturation limits. The wave based control strategy launches a wave equal to half the reference signal $\theta_{ref}$. The measured values of $\theta$ and $M$ are then used to calculate the returning wave component at the actuator which can be calculated as:

$$B_0 = H_0 \left( \frac{\theta - P_0 M}{1 - P_0 Q_0} \right)$$  \hspace{1cm} (38)$$

The actuator is then moved to match this returning wave component and thereby absorb it. When the absorbing is finished the system will have been displaced by twice the specified launch wave, i.e. will be at the reference displacement.

4.2 Time-Optimal Controller

For comparison, the torque-limited bang-bang solution for a rest-to-rest maneuver of a rigid rocket was calculated. With the slosh pendulum frozen in position the moment of inertia of the rocket body about the overall mass centre is given by:

$$I_{\text{rigid}} = I + \frac{mm_f (b-a)^2}{(m+m_f)}$$  \hspace{1cm} (39)$$
The switching time for a rest to rest maneuver is:

\[ t_s = \sqrt{\frac{\theta_{\text{ref}} I_{\text{rigid}}}{M_{\text{max}}}} \]  

(40)

where \( M_{\text{max}} \) is the maximum torque. Then the control input for the bang-bang maneuver beginning at \( t = t_0 \) is:

\[
M = \begin{cases} 
0 & t < t_0 \\
M_{\text{max}} & t_0 < t < t_0 + t_s \\
-M_{\text{max}} & t_s < t < t_0 + 2t_s \\
0 & t > t_0 + 2t_s 
\end{cases}
\]  

(41)

5 RESULTS

6 Simulation of a real rocket

The wave-based controller was tested by numerical simulation. Suitable parameters for the presented rocket model were chosen to represent AVUM, upper stage of the European Vega launcher [9] (table 2). The included slosh pendulum represents the primary sloshing mode for the rocket’s fuel tank when half full. The saturation torque \( M_{\text{max}} \) was calculated from the maximum gimbal angle of the AVUM engine. The values chosen for notional mass and spring stiffness were \( m_0 = m_1 \) and \( k_0 = k_1 \times (m_1/m_2) \), however the choice for these parameters is arbitrary to some degree and a range of values will give a good control response. Results are shown in figure 7 for a five degree step change in commanded pitch angle \( \theta_{\text{ref}} \). The wave-based controller is compared to the torque-limited time-optimal solution for the rigidized rocket.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>( m )</td>
<td>2105</td>
<td>kg</td>
</tr>
<tr>
<td>( m_f )</td>
<td>88</td>
<td>kg</td>
</tr>
<tr>
<td>( I )</td>
<td>1883</td>
<td>kg m^2</td>
</tr>
<tr>
<td>( a )</td>
<td>0.53</td>
<td>m</td>
</tr>
<tr>
<td>( b )</td>
<td>-1.43</td>
<td>m</td>
</tr>
<tr>
<td>( F )</td>
<td>2450</td>
<td>N</td>
</tr>
<tr>
<td>( M_{\text{max}} )</td>
<td>931</td>
<td>N m</td>
</tr>
</tbody>
</table>

7 DISCUSSION

It can be seen that the fuel slosh dynamics cause the open-loop time optimal controller to land off target and drift away from the target over time. The fuel sloshing persists for long times in the absence of damping in the model. The wave-based controller lands on target and suppresses the sloshing motion. However the sloshing persists for several oscillations. The reason for this is the non-uniformity of the system, i.e. unequal inertias \( m_1 \) and \( m_2 \). In this case \( m_1 \) is much less than \( m_2 \). From a wave perspective there is a change in wave impedance between the two different masses and some waves become trapped on the right hand side of this boundary. For this reason the actuator only absorbs a fraction of the motion on each oscillation cycle, but over several cycles can absorb it all. When the ratio of inertias \( m_1/m_2 \) is much less than one, the effect of the pendulum on the body is much reduced and so it takes longer to fully suppress sloshing motions. This is acceptable, however, because in this case, by definition, the sloshing does not cause a major problem for the rocket controller. On the other hand, the control challenge is greatest when the fluid inertia ratio is large, and this is precisely when the new strategy delivers much
improved performance. An interesting avenue of future research is developing wave-models and controllers which take into account the non-uniformity of the system to be controlled. A clear advantage of wave-based control is that all measuring is done at the actuator, in this case the rocket body, so no measurement of the pendulum states is necessary, which is a significant bonus given the challenge of measuring or modelling them in a real rocket. Future research includes making the controller robust to external disturbances such as aerodynamic, stage separation or gravity forces; considering non-ideal actuator and sensor behaviour; including multiple slosh pendulums representing either multiple fuel tanks or multiple modes of sloshing in a single tank; and extending the analysis to a 6-DOF model where roll, pitch and yaw must be simultaneously controlled.

REFERENCES


Figure 7. Pitch $\theta$ and pendulum angle $\phi$ for a 5 degree step change in $\theta_{ref}$. 
Comparison of control and optimization approaches for trajectory tracking in the forward dynamic simulation of biomechanical multibody systems

Álvaro Noriega*, Urbano Lugrís#, Javier Cuadrado#

*Department of Construction and Manufacturing Engineering
University of Oviedo
Campus de Gijón, C/Pedro Puig
Adam, Edf. Dep. Oeste, Módulo 5, 33203 Gijón, Spain
noriegalvaro@uniovi.es

#Laboratory of Mechanical Engineering
University of La Coruña
Escuela Politécnica Superior, Mendizábal s/n, 15403 Ferrol, Spain
ulugris@udc.es

ABSTRACT
In the study of biomechanical systems, it is common to have a parametric model of a multibody system which should perform a certain movement, usually defined by a motion capture of the system in real conditions. From these capture data, the inverse dynamic analysis provides a solution for drive torques that, when introduced into a forward dynamic analysis, generates a movement which is different from the one previously captured. In this paper, a simple multibody model and a movement are proposed to benchmark two approaches for the trajectory tracking problem: a control approach featuring a PD control with computed feedforward and PD feedback, and an optimization approach based on the parameterized histories of drive torques with and without computed feedforward. Results show that the proposed multibody model generates the same problems than more complex models, despite its apparent simplicity. It is also shown that the PD control with computed feedforward is faster and more accurate than any of the tested optimization methods.

Keywords: Biomechanics, Multibody system, Forward dynamics simulation, Control, Optimization.

1 INTRODUCTION
In the study of biomechanical systems and, more specifically, in the study of human gait, it is common to have a model of multibody dynamics for the individual whose motion is to be analyzed. This model can have more or less detail, but full fidelity is impossible since many biomechanical parameters are not directly measurable and must be estimated by means of scaling of tabulated data obtained from other analyzed individuals [1, 2].

It is also common to have an optical motion capture system as indicated in [3]. This system captures the position histories of a group of markers that are assumed rigidly attached to the bones of the individual. Data from this capture are filtered to remove noise and, from them, the coordinates that define the instantaneous position of the individual are calculated.

These histories of coordinates have no kinematic consistency since markers, in reality, are not rigidly attached to the bones and there may exist inaccuracies in the measurement. Therefore, data from these histories must undergo a process to ensure their kinematic consistency [4].

Once the histories of the coordinates that define the position of the biomechanical model are available, the histories of the degrees of freedom of the model (usually associated with rotations in joints) can be obtained.

After that, an approximation of these histories is carried out, for instance, by cubic splines. The histories of the coordinates can be differentiated with respect to time with different numerical techniques to obtain the velocity and acceleration, but the use of cubic splines is advantageous since it allows to implement the analytical differentiation very easily [3].
Then, if an inverse dynamic analysis is made with the histories of movement in the degrees of freedom calculated with the previously indicated procedure, the drive torques in the joints can be obtained.

To validate the histories of the drive torques so obtained, they can be introduced in the biomechanical model and perform a forward dynamic analysis to check whether the original captured movement is recovered. Conversely to what could be expected, it is found that the resulting gait is unstable [3].

The origin of this divergence of behaviours can be multiple. Possible errors in measurements, filtering, the process to ensure the kinematic consistency or the use of splines can lead to minor differences between the actual data and the data used to solve the inverse dynamic analysis prior to forward dynamic analysis. These differences are amplified during the process of integration of the differential equations of the model. The differences may also come from the mathematical modelling of real contacts between the biomechanical system and the environment. Finally, the integration process itself assumes that functions are continuous, which can lead to differences in points of rapid change or even discontinuity of these functions.

The interest of using forward dynamic analysis in simulation and control of biomechanical systems is due to the fact that it naturally enables the introduction of muscle models thus leading to a more realistic behaviour. To address the trajectory tracking problem various approaches have been proposed in the literature [5], being the most important ones control and optimization. Both have advantages and disadvantages, as indicated by Xiang [5], apparently being optimization a better choice for motion prediction, while control is preferable to track a known movement. However, it is interesting to quantify these advantages and disadvantages in a practical way.

To that end, the use of a simple multibody model with a single degree of freedom which allows fast simulation, but where typical problems of complex multibody models can occur, is proposed. A predefined movement is set to avoid the process of motion capture, filtering and kinematic consistency, which adds nothing to this research. The objective is that the multibody model tracks the predefined movement as accurately as possible within a forward dynamic analysis.

Two approaches are proposed to do this. On the one hand, a forward dynamic analysis with PD control and computed feedforward from the solution obtained in the inverse dynamic analysis is performed. On the other hand, a forward dynamic optimization where the design variables are the histories of the drive torques is carried out. These histories will be approximated by three techniques: Artificial Neural Networks (ANN), cubic splines (CS) and parametric functions (PF). The optimization is performed with and without computed feedforward and several optimization algorithms are used to solve the problem.

The performance of the different resulting methods is measured by means of the following three indicators:

- Root of mean squared error with respect to the desired rotation ($RMSE_{rot}$).
- Root of mean squared error with respect to the drive torque obtained in the inverse dynamic analysis ($RMSE_{torque}$).
- Runtime ($t$).

The paper is organized as follows. The second section describes the multibody model and the formulation for inverse dynamic analysis. The third section shows the control approach through a PD controller with computed feedforward. The fourth section presents the optimization approach with and without computed feedforward. The fifth section compares the results obtained by the different methods. Finally, the sixth section gathers the conclusions.
A planar forearm multibody model with 1 degree of freedom in the elbow is proposed as case study. The schematic is shown in Figure 1.

The kinematic model is implemented by means of the mixed coordinates [6],

\[ \mathbf{q} = [x_2 \quad y_2 \quad \theta]^T \]  \hspace{1cm} (1)

with the elbow rotation as independent coordinate,

\[ \mathbf{z} = [\theta]^T \]  \hspace{1cm} (2)

The formulation to solve the inverse dynamic analysis at each instant of time and to calculate the drive torque at the elbow (\( M_{\text{elbow}} \)) is defined by expression (3) and its terms are explained in (4). Constants are indicated in Table 1.

\[ \mathbf{Q}_m = \mathbf{R}^T \cdot (\mathbf{M} \cdot \ddot{\mathbf{q}} - \mathbf{Q}_{\text{ext}}) \]  \hspace{1cm} (3)

\[
\mathbf{M} = \begin{bmatrix}
\frac{I_1}{L_{12}} & 0 & 0 \\
0 & \frac{I_1}{L_{12}} & 0 \\
0 & 0 & 0
\end{bmatrix}
\]

\[
\mathbf{Q}_{\text{ext}} = \begin{bmatrix}
0 & -F_{\text{ext}} \\
0 & -m \cdot g
\end{bmatrix} + \mathbf{C}_G^T \cdot \begin{bmatrix}
0 \\
0
\end{bmatrix}
\]

\[
\mathbf{Q}_m = \begin{bmatrix}
0 \\
0 \\
M_{\text{elbow}}
\end{bmatrix}
\]  \hspace{1cm} (4)

\[
\mathbf{C}_G = \frac{L_{G1}}{L_{12}} \cdot \begin{bmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 0
\end{bmatrix}
\]

\[ \mathbf{R} = \dot{\mathbf{q}}(z(\theta), \ddot{z}(\theta, 1)) \]

Table 1. Constant values.
The simulation is performed in the time range \([0,1]\) s, extracting values every 0.001 seconds.

The desired movement in the elbow is defined as is shown in (5):

\[
\begin{align*}
t^* (t) &= a_1 \cdot t + (1-a_1) \cdot t^2 \\
\theta (t^*) &= L \cdot \left( t^* - \frac{1}{2 \cdot \pi} \cdot \sin \left( 2 \cdot \pi \cdot t^* \right) - 1 \right) \\
\dot{\theta} (t^*) &= L \cdot \left( 1 - \cos \left( 2 \cdot \pi \cdot t^* \right) \right) \\
\ddot{\theta} (t^*) &= 2 \cdot \pi \cdot L \cdot \sin \left( 2 \cdot \pi \cdot t^* \right)
\end{align*}
\]

Being \(a_1 = 0.05\) and \(L = \pi/4\).

These expressions allow to define the vector of independent coordinates and its derivatives as,

\[
z = [\theta(t)]^T \quad \dot{z} = [\dot{\theta}(t)]^T \quad \ddot{z} = [\ddot{\theta}(t)]^T
\]

The equations to calculate the position, velocity and acceleration of the dependent coordinates are,

\[
\begin{align*}
x_2 &= L_{12} \cdot \cos (\theta) \\
y_2 &= L_{12} \cdot \sin (\theta) \\
x_2 &= -L_{12} \cdot \sin (\theta) \cdot \dot{\theta} \\
y_2 &= L_{12} \cdot \cos (\theta) \cdot \dot{\theta} \\
x_2 &= -L_{12} \cdot \left( \cos (\theta) \cdot \dot{\theta}^2 + \sin (\theta) \cdot \ddot{\theta} \right) \\
y_2 &= L_{12} \cdot \left( -\sin (\theta) \cdot \dot{\theta}^2 + \cos (\theta) \cdot \ddot{\theta} \right)
\end{align*}
\]

These expressions allow to define the vector of dependent coordinates and its derivatives,

\[
q = \begin{bmatrix} x_2 & y_2 & \theta \end{bmatrix}^T = q(z) \quad \dot{q} = \begin{bmatrix} \dot{x}_2 & \dot{y}_2 & \dot{\theta} \end{bmatrix}^T = q(z, \dot{z})
\]

\[
\ddot{q} = \begin{bmatrix} \ddot{x}_2 & \ddot{y}_2 & \ddot{\theta} \end{bmatrix}^T = \ddot{q}(z, \dot{z}, \ddot{z})
\]

In Figure 2, the forearm rotation and the drive torque in the elbow obtained by inverse dynamic simulation are shown.
Figure 2. Forearm rotation and drive torque in the elbow obtained from inverse dynamic simulation.

3 FORWARD DYNAMIC ANALYSIS WITH PD CONTROL AND COMPUTED FEEDFORWARD

For the coordinates defined in (1), the set of kinematic constraints is specified in (9), the Jacobian matrix of the kinematic constraints is shown in (10), and its time derivative in (11).

\[
\Phi(q) = \begin{bmatrix}
(x_2 - x_1)^2 + (y_2 - y_1)^2 - L_{12}^2 \\
(y_2 - y_1) - L_{12} \cdot \sin(\theta)
\end{bmatrix} = 0
\] (9)

\[
\Phi_q(q) = \begin{bmatrix}
2 \cdot (x_2 - x_1) & 2 \cdot (y_2 - y_1) & 0 \\
0 & 1 & -L_{12} \cdot \cos(\theta)
\end{bmatrix}
\] (10)

\[
\dot{\Phi}_q(q) = \begin{bmatrix}
2 \cdot \dot{x}_2 & 2 \cdot \dot{y}_2 & 0 \\
0 & 0 & L_{12} \cdot \sin(\theta) \cdot \dot{\theta}
\end{bmatrix}
\] (11)

For the forward dynamic analysis, a penalty formulation is used [6]. It is defined by the following system of equations,

\[
\left( M + \alpha \cdot \Phi_q^T \cdot \Phi_q \right) \cdot \ddot{q} = Q - \alpha \cdot \Phi_q^T \cdot \left( \dot{\Phi}_q \cdot q + 2 \cdot \dot{\xi} \cdot \omega \cdot \Phi + \omega^2 \cdot \Phi \right)
\] (12)

The vector of generalized forces is described in (13) and the term due to the PD control is shown in (14).

\[
Q = Q_{ext} + Q_m + Q_{control}
\] (13)

\[
Q_{control} = K_p \cdot \begin{bmatrix}
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 1
\end{bmatrix} \cdot \left( q_{desired} - q(z) \right) + K_D \cdot \begin{bmatrix}
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 1
\end{bmatrix} \cdot \left( \dot{q}_{desired} - \dot{q}(z, \dot{z}) \right)
\] (14)
4 FORWARD DYNAMIC OPTIMIZATION

4.1 Without computed feedforward

The forward dynamics formulation for the forearm, with the drive torque defined by an Artificial Neural Network (ANN), a Cubic Spline (CS) or a Parametric Function (PF), is the one indicated in Section 3, except that the vector of generalized forces is the expression given in (15), where the term corresponding to the additional torque is shown in (16).

\[ \mathbf{Q} = \mathbf{Q}_{ext} + \mathbf{Q}_{\text{motor}} \quad (15) \]

\[ \mathbf{Q}_{\text{motor}} = \begin{bmatrix} 0 & 0 & \text{ANN}(t, \mathbf{p}_{\text{ANN}}) \text{ or } \text{CS}(t, \mathbf{p}_{\text{CS}}) \text{ or } \text{PF}(t, \mathbf{p}_{\text{PF}}) \end{bmatrix}^T \quad (16) \]

being

\( \text{ANN}(t, \mathbf{p}_{\text{ANN}}) \) a parametric Artificial Neural Network which describes the history of drive torque in time;

\( \text{CS}(t, \mathbf{p}_{\text{CS}}) \) a parametric Cubic Spline which describes the history of drive torque in time;

\( \text{PF}(t, \mathbf{p}_{\text{PF}}) \) a Parametric Function which describes the history of drive torque in time.

If the history of the torque applied to the elbow is defined by an ANN, the artificial network has the structure shown in Figure 3.

![Figure 3. Structure of the ANN used.](image)

ANN seems a good choice because of its flexibility to precisely adapt to non-regular shapes. The ANN is, in fact, a parametric model where the parameters are the weights of the synapses that connect neurons and the bias in these neurons. For the ANN shown in Figure 3, the number of parameters is \( 3 \cdot n_h + 1 \) if the number of neurons in the hidden layer is \( n_h \).

The number of neurons in the hidden layer determines the complexity that the torque history can have. The higher the number, the greater the complexity. The optimal number of hidden layer neurons generates a fit to the training data with sufficient accuracy while minimizing the number of neurons (which, in turn, minimizes the number of parameters), and is determined with the method shown in [7]. In this case, the training data come from the inverse dynamic
analysis, because it is assumed that the optimum history of drive torque for the forward dynamic analysis is quite similar. Therefore, a prior training of the ANN must be made with these data to determine the optimal number of neurons for the hidden layer, and the weights of synapses and bias for this optimal number of neurons. These values of the parameters for the ANN serve as initial guess for the optimization problem to be solved later. This process has been carried out and the optimal number of 4 neurons in the hidden layer has been obtained. The weights of synapses and bias that adjust the ANN to the results of the inverse dynamic simulation have also been obtained.

In case the history of the drive torque applied to the elbow is defined by a cubic spline, a set of time instants (here, every 0.1 s) are defined, being the parameters of the spline the values of torque at those instants, as shown in Figure 4.

![Figure 4. Cubic spline to describe the history of the drive torque at the elbow.](image)

A cubic spline is very flexible to adapt itself to non-regular shapes with precision, but this adaptation is achieved by increasing the density of points that define the history, which means an increase in the number of required parameters.

Finally, if the history of drive torque in the elbow is defined by a parametric function, its structure must be defined as well as their parameters and the ranges in which they can move while maintaining the validity of the function. In this study, many postulated functions governed by simple parameters and sums of such functions have been tried, but to determine a parametric function that adapts itself with enough accuracy to the history of the torque obtained from the inverse dynamic simulation is almost impossible. Furthermore, the postulated function would only adapt to a very specific shape. If the movement changes, the drive torque also changes and then it is very probable that the former parametric function does not serve. For this reason, the PF without computed feedforward is not included in the benchmark.

Once the torque history is available by any of the described methods, the forward dynamic formulation explained in (12) can be implemented and the simulation of the movement with a set of parameters (\( \mathbf{p}_{\text{ANN}} \), \( \mathbf{p}_{\text{CS}} \) or \( \mathbf{p}_{\text{PF}} \)) carried out. The movement obtained from this simulation can be compared with the desired movement in order to obtain \( \text{RMSE}_{\text{rot}} \). Then, the following optimization problem can be stated,

\[
\min f(\mathbf{p}) = \min \text{RMSE}_{\text{rot}}(\mathbf{p})
\]

Being \( f(\mathbf{p}) \) the objective function that calculates the error indicated depending on the parameters that define the history of elbow torque (\( \mathbf{p} \)). This function internally makes the forward dynamic simulation of the motion with the formulation indicated in (12) and with the history of elbow drive torque defined by the ANN, the CS or the PF.
As the optimization algorithm requires that the objective function be surjective, the handling of cases where the integrator cannot complete the forward dynamic simulation in the complete range of time must be considered. To that end, the objective function is modified by adding a term which is proportional to the not completed range of time, \(1 - t_{\text{end}}\), as shown in (18). Thus, sets of parameters that generate unstable simulations are penalized, being the penalty proportional to the degree of instability.

\[
\min f^* (p) = \min \left( \text{RMSE}_{\text{rot}} (p) + \omega_{\text{penal}} \cdot (1 - t_{\text{end}}) \right)
\]

To select the optimization algorithm, the features of the problem and of the objective function must be taken into account:

- It is a minimization problem of a single objective.
- It has a large number of continuous variables.
- The derivative of the objective function with respect to the variables is unknown.
- It is unknown whether there will be one or more optima, and their relative values are also unknown.
- An approximation of the optimal solution is known (from inverse dynamic analysis).

Based on these features, two algorithms of different types are selected. On the one hand, evolutionary strategy CMA-ES [8] is chosen because it natively works with continuous variables, besides showing good performance in functions with several optima. On the other hand, a BFGS quasi-Newton method is selected, for which derivatives are numerically estimated. Both algorithms typically require few objective function evaluations. This is an interesting property to keep runtime within acceptable values.

Finally, the weight \(\omega_{\text{penal}}\), used to modulate the relevance of the set of parameters being capable of generating a stable simulation with the integrator used, is set to 1.

4.2 With computed feedforward

The formulation with computed feedforward is the same indicated in Section 3, adding the feedforward term to the vector of generalized forces as shown in (19).

\[
Q = Q_{\text{ext}} + Q_m + Q_{\text{motor}}
\]

This approach is operationally similar to that described in Section 4.1 except that, in this case, the elbow drive torque is added to the one coming from inverse dynamic analysis. This makes that, most of the time, the value of the additional torque should tend to 0 and it only grows when there are significant variations in the error in position or its derivatives. Consequently, the initial guess for the parameters of the ANN and the CS is 0.

For the case of PF, the parametric function proposed in [2] is considered. This function has \(3 \cdot n_i\) parameters, being \(n_i\) the number of terms. Its expression is shown in (20).

\[
PF(t, p_{PF}) = \sum_i A_i \cdot e^{-C_i (t-T_i)}
\]

Being \(p_{PF} = [A_i \ C_i \ T_i \ \cdots]\)
5 RESULTS

The code has been programmed in MATLAB® on a PC with Intel® Core Duo 2.67 GHz, 3.5 GB RAM and Windows 7. The inverse dynamic simulation has a computation time of 0.12 s. Forward dynamic simulation with PD control and computed feedback was performed with a gain of $K_p = 80$ and $K_D = 0.3$ which have been adjusted by trial and error method. Simulation time was 0.075s and generates errors $RMSE_{\text{rot}} = 0.116$ and $RMSE_{\text{torque}} = 0.149$.

The forward dynamic optimization poses multiple options whose results are shown in Tables 2 and 3. Table 2 shows the case without computed feedforward and Table 3 shows the case with computed feedforward. The optimization problems have been solved allowing a maximum of 200 evaluations of the objective function in the optimization algorithms.

<table>
<thead>
<tr>
<th>History of drive torque defined by …</th>
<th>Optimization method used</th>
<th>CMA-ES</th>
<th>BFGS</th>
<th>Quasi-Newton</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANN (4 hidden neurons)</td>
<td>$RMSE_{\text{rot}} = 1.250$</td>
<td></td>
<td></td>
<td>$RMSE_{\text{rot}} = 0.310$</td>
</tr>
<tr>
<td></td>
<td>$RMSE_{\text{torque}} = 0.149$</td>
<td>411.56s</td>
<td></td>
<td>$RMSE_{\text{torque}} = 0.121$</td>
</tr>
<tr>
<td></td>
<td>$t = 411.56s$</td>
<td>449.37s</td>
<td></td>
<td>$t = 449.37s$</td>
</tr>
<tr>
<td>CS (11 points)</td>
<td>$RMSE_{\text{rot}} = 0.558$</td>
<td></td>
<td></td>
<td>$RMSE_{\text{rot}} = 0.197$</td>
</tr>
<tr>
<td></td>
<td>$RMSE_{\text{torque}} = 0.118$</td>
<td>29.03s</td>
<td></td>
<td>$RMSE_{\text{torque}} = 0.116$</td>
</tr>
<tr>
<td></td>
<td>$t = 29.03s$</td>
<td>41.59s</td>
<td></td>
<td>$t = 41.59s$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>History of drive torque defined by …</th>
<th>Optimization method used</th>
<th>CMA-ES</th>
<th>BFGS</th>
<th>Quasi-Newton</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANN (4 hidden neurons)</td>
<td>$RMSE_{\text{rot}} = 27.383$</td>
<td></td>
<td></td>
<td>$RMSE_{\text{rot}} = 2.079$</td>
</tr>
<tr>
<td></td>
<td>$RMSE_{\text{torque}} = 0.366$</td>
<td>2439.15s</td>
<td></td>
<td>$RMSE_{\text{torque}} = 0.009$</td>
</tr>
<tr>
<td></td>
<td>$t = 2439.15s$</td>
<td>1482.89s</td>
<td></td>
<td>$t = 1482.89s$</td>
</tr>
<tr>
<td>CS (11 points)</td>
<td>$RMSE_{\text{rot}} = 1.284$</td>
<td></td>
<td></td>
<td>$RMSE_{\text{rot}} = 0.696$</td>
</tr>
<tr>
<td></td>
<td>$RMSE_{\text{torque}} = 0.105$</td>
<td>35.53s</td>
<td></td>
<td>$RMSE_{\text{torque}} = 0.046$</td>
</tr>
<tr>
<td></td>
<td>$t = 35.53s$</td>
<td>35.67s</td>
<td></td>
<td>$t = 35.67s$</td>
</tr>
<tr>
<td>PF (4 terms)</td>
<td>$RMSE_{\text{rot}} = 1.829$</td>
<td></td>
<td></td>
<td>$RMSE_{\text{rot}} = 1.816$</td>
</tr>
<tr>
<td></td>
<td>$RMSE_{\text{torque}} = 0.012$</td>
<td>24.11s</td>
<td></td>
<td>$RMSE_{\text{torque}} = 0.095$</td>
</tr>
<tr>
<td></td>
<td>$t = 24.11s$</td>
<td>17.09s</td>
<td></td>
<td>$t = 17.09s$</td>
</tr>
</tbody>
</table>

In Figure 5, we can appreciate the discrepancy between the desired motion and the results provided by control-based approach and the optimization-based without feedforward approaches. In Figure 6, we can see the same comparison made with the optimization-based with feedforward approaches.
In general, optimization-based approaches work better without feedforward. Furthermore, the results show that the optimal solution is very sensitive to the optimization method used, with the BFGS quasi-Newton method obtaining the best results in all cases.

6 CONCLUSIONS

First, it has been shown that despite the simplicity of the proposed multibody model, it can generate the same problems as more complex models in the forward dynamic simulation. Several approaches have been proposed to parameterize the history of drive torque in a flexible way. The control-based approach is almost 3 or 4 orders of magnitude faster and more accurate than the optimization-based approaches.
Optimization-based approaches involve a high number of function evaluations, i.e. forward dynamic simulations. Moreover, convergence to a solution with an accuracy comparable to that obtained with the PD controller and within a reasonable time cannot be ensured. For this reason, the computational effort was limited in all the optimization algorithms.

In optimization-based approaches, the CS with and without computed feedforward has shown the best behaviour. Surprisingly, optimization-based approaches without feedforward have better behaviour than their counterparts with computed feedforward.

Regarding efficiency, the ANN is slower than the other two methods due to two reasons. First, the ANN needs a long time to be evaluated (around tenths of seconds), and this process must be repeated at every simulation instant (every 0.001s) or even more, depending on the integrator.

Second, the drive torque history may include sudden variations that hinder the integration of the equations of motion, making the integrator to reduce the time-step size.

Finally, the results of optimization-based approaches have shown a greater or lesser dependence on the optimization methods used with them and even on the parameters of these methods.

ACKNOWLEDGEMENTS

The support of this work by the Spanish Ministry of Economy and Competitiveness (MINECO) under project DPI2012-38331-C03-01, co-financed by the European Union through EFRD funds, is greatly acknowledged.

REFERENCES


ABSTRACT

A prosthetic swing-phase control mechanism simulates the action of the upper leg musculature to aid in increased gait function. More specifically, swing-phase control mechanisms limit the maximum knee flexion and allow the shank to smoothly decelerate into full knee extension, without excessive impact. In this work, a magnetorheological (MR) damper is designed with the objective of controlling swing-phase damping in an above-knee prosthesis. A parametric model, the modified Bouc-Wen model, is used to represent the highly nonlinear dynamic properties of the MR damper. Based on this model, twelve control parameters that govern the hysteretic force and displacement of the damper have been identified. The parameters of the damper are determined through optimization of the prosthesis knee angle with a desired knee angle trajectory obtained from experimental data in normal level walking. Experimental data of thigh and hip motions are introduced as input into a dynamic system to find out a set of control parameters. A computer simulation is carried out. Comparison of the desired knee angle with the knee angle obtained from control parameters of the designed MR damper shows the effectiveness of the present design. Also, using the optimal control parameters, knee angle trajectories at zero and at lowered input currents, representing circumstances when the battery turns off and the power supply is reduced respectively, have been shown. Moreover, conditions of knee angle and shank velocity at the end of swing phase have been checked. The results obtained show a satisfactory performance of the system.

Keywords: Swing phase control, Magnetorheological (MR) damper, Modified Bouc-Wen, Single-axis knee, Prosthetics, Above-knee amputee

1 INTRODUCTION

Human gait is characterized by periodic repetitions of two phases: a stance phase in which a foot is in contact with the ground, followed by a swing phase in which the lower limb swings through after toe-off. The functional necessities of above-knee prostheses are to provide knee stability during the stance phase and damping during the swing phase. Therefore, the prosthetic knee is a key component of above-knee lower limb prostheses, possibly, the most complex. The ideal prosthesis should mimic the alignment and gait characteristics of the normal limb during each of the phases of the gait cycle and must provide safety, stability, reliable support when standing, smooth controlled motion when walking and permit unrestricted movement for sitting, bending and kneeling [1]. Most conventional above-knee prostheses are based on passive mechanisms with constant mechanical properties such as friction, spring and damping coefficients. However, research has shown that the swing phase of human walking is not a passive movement [2]. Thus, passive designs are insufficient for mimicking natural gait due to lack of active knee joint control. On the other hand, active prostheses, such as those with hydraulic actuation (C-Leg, Otto Bock) and a magnetorheological (MR) brake (Rheo Knee, Össur) can produce a gait similar to that of normal persons effectively, but are expensive, heavy, provide inadequate response when the battery runs out and consume significant energy. Semi-active type prostheses can give enhanced knee stability, adaptation to changing environment and ability to change walking speed. Such designs improve the gait symmetry, which leads to a reduction in degenerative musculoskeletal changes and energy consumption for the amputee [3, 4]. Moreover, as shown in Figure 1 plotted based on data from [5], whenever knee angular velocity and knee torque have the same sign, knee muscles act like a passive
damper and at other times, they act as actuators, hence the knee requires both active and passive damping forces within the swing phase. Therefore, the semi-active type prostheses have been developed as they appear to have desirable features of passive and active systems.

MR fluid damper is a semi-active device that is widely used in variable damping knees [6-8]. MR fluid is a suspension of micrometer-sized magnetic particles in a carrier fluid, which is usually a type of oil. In the absence of an applied field, the particles are distributed randomly and the fluid exhibits quasi-Newtonian behavior. When the MR fluid is subjected to a magnetic field, the particles become magnetized (induced dipoles) and they start to behave like tiny magnets. The interaction between the resulting induced dipoles causes the particles to aggregate and form fibrous structures within the carrier liquid (chains or column like structures), changing the rheology of the MR fluid to a near solid state. These chain-like structures restrict the flow of the MR fluid, thereby increasing the viscous characteristics of the suspension. The mechanical energy needed to yield these chain-like structures increases nonlinearly with an increase of the applied magnetic field, resulting in a field-dependent yield stress. The process is fully variable and reversible. By controlling the strength of the magnetic field, the shear strength of the MR fluid can be altered, so that resistance to the MR flow can be varied. Some researchers have studied reliability of MR damper application in the rehabilitation area [9-13].

Since the parameters of prosthetic knee have an essential effect on its function during swing phase, they should be studied to make a prosthetic gait as close as possible to normal one. Some studies have engaged in mathematical simulation of these parameters. Tahani and Karimi [14] proposed a simple dynamic model of prosthesis using torsional spring and optimized control parameters for swing phase motion. Suzuki [15] performed dynamic optimization of a musculoskeletal model of residual limb to get optimal knee joint friction value of passive prosthetic knee such that muscle metabolic energy expenditure is minimized during swing phase. Hong-Liu et al. [16] developed a dynamic model of the swing phase for an intelligent prosthetic leg (IPL) system, based on the control parameters of a nonlinear hydraulic damper, to identify the dynamic interaction between the swing speed and the opening of needle valve at the damper.
Most of the researchers have shown the application of MR damper to prosthetic knee based on available dampers in the market. However, in order to achieve better performance, it has to be designed specifically for prosthetic knee use. Therefore, in this work, it is aimed at producing a platform of designing MR damper for semi-active type above-knee prostheses. In the present study, a simple mechanism of single-axis knee joint using MR damper is considered during the swing phase. The damper is designed with the objective of controlling swing-phase damping in above-knee prosthesis. Control parameters are optimized so as to achieve an optimal track of the swing phase trajectory of a semi-active prosthetic knee.

2 SINGLE AXIS PROSTHETIC KNEE INCORPORATING MR DAMPER

The single-axis knee in focus is a mechanism with one degree of freedom which is to be attached to the socket housing the residual limb of a transfemoral amputee. In designing the controller for this knee, the thigh motion from experimental data is provided to the dynamic system model to achieve a desired shank motion during the swing phase of the walking cycle. The motion is accomplished by incorporating and controlling the MR damper through a controlled electric current.

2.1 Dynamic System Modelling

The knee modeled as a single-axis knee with an MR damper as shown in Figure 1. The ankle is assumed to be rigid. The swing leg of the amputee is modeled as a two-link rigid body chain representing the thigh and the shank in sagittal plane motion. In the figure, subscripts 1 and 2 represent the thigh and shank respectively, \( m_i \) are the masses, \( a_i \) are the distances of the mass centres from the respective proximal joints, \( I_i \) are the moments of inertia, \( l_i \) are lengths and \( \theta_i \) are the absolute angles of thigh and shank from the horizontal; \( \theta_t \) and \( \theta_s \) are the corresponding absolute angles of thigh and shank respectively from the vertical, \( s \) is the offset between the knee centre and location of attachment of MR damper piston on the thigh, \( \theta_k = \theta_t + \theta_s \) is the knee angle, \( l_d \) is the length of the MR damper and \( b \) is the distance between the knee centre and location of the MR damper attachment on the shank. \( x_h \) is the horizontal movement of hip, and \( y_h \) is the vertical movement of hip and \( T_1 \) is hip torque.

Assuming that at each joint there is no friction and using Lagrange’s formulation, one can develop the equation of motion in the following form:

\[
\begin{align*}
D(\theta)\ddot{\theta} + C(\theta, \dot{\theta}) + G(\theta) &= \Gamma \\
D(\theta) &= \begin{bmatrix}
m_1 a_1^2 + l_1 + m_2 l_1^2 & -m_2 l_1 a_2 \cos(\theta_1 + \theta_s) \\
-m_2 l_1 a_2 \cos(\theta_1 + \theta_s) & m_2 a_2^2 + l_2
\end{bmatrix} \\
C(\theta, \dot{\theta}) &= \begin{bmatrix}
m_2 l_1 a_2 (\dot{\theta}_s)^2 \sin(\theta_1 + \theta_s) \\
m_2 l_1 a_2 (\dot{\theta}_1)^2 \sin(\theta_1 + \theta_s)
\end{bmatrix} \\
G(\theta) &= \begin{bmatrix}
(m_2 l_1 + m_2 a_2)(g \sin \theta_1 + \dot{x}_h \cos \theta_1 + \dot{y}_h \sin \theta_1) \\
m_2 a_2 (g \sin \theta_s - \dot{x}_h \cos \theta_s - \dot{y}_h \sin \theta_s)
\end{bmatrix} \\
\Gamma &= \begin{bmatrix}
T_1 + F_d b \sin(\theta_s - \beta) \\
-F_d b \sin(\theta_s - \beta)
\end{bmatrix}
\end{align*}
\]

The MR damper’s length is a variable length, \( l_d \) and the damper’s upper part is connected to the lower perpendicular posterior extension of the thigh through a pin joint at length \( s \) from the thigh-knee line, and the damper’s lower part is connected with the shank at an offset \( b \) from the knee on knee-shank line. Considering the standard swing motion position adopted from [5],
damping force $F_d$ lies along the line connecting the two pin joints of the damper making angle $\beta$ from the vertical. Therefore, considering Figure 2, $\beta$ can be geometrically determined to be:

$$\beta = \cos^{-1} \left( \frac{1 - \frac{b}{l} \sin \theta_k}{\frac{1}{1 + \left( \frac{b}{l} \right)^2} - 2 \left( \frac{b}{l} \right) \sin \theta_k} \right) - \theta_t - 90^\circ \tag{2}$$

Figure 1. The amputee's swing leg model

Figure 2. Damper force resolution
In the computation of dynamic equations of motion, the hip torque generated and the thigh angle of normal person are known inputs. Therefore, in controlling the knee angle, it is the second row of equation 1 that needs to be considered for further computation. And hence, after simplifying the equation, equation 3 has been developed.

\[
L \ddot{\theta}_s = F_d \sin(\theta_s - \beta) + N \cos \theta_s + P \sin \theta_s \tag{3}
\]

where \( L = -\left(\frac{m_2 a_2^2 + l_2}{m_2 a_2}\right) \)

\[
N = (-l_1 \cos \theta_t \ddot{\theta}_t + l_1 \sin \theta_t (\ddot{\theta}_t)^2 - \dddot{x}_h)
\]

\[
P = \left( l_1 \sin \theta_t \ddot{\theta}_t + l_1 \cos \theta_t (\ddot{\theta}_t)^2 + g - \dddot{x}_h \right) \quad \text{and} \quad M = \left( \frac{b}{m_2 a_2}\right)
\]

### 2.2 MR Damper Model

Due to the highly nonlinear dynamic properties of the MR damper, system identification is mandatory for accurate control. In this work, the most commonly used parametric model, modified Bouc-Wen, proposed by Spencer et al. [17] and shown in Figure 3, has been used.

![Figure 3. Modified Bouc-Wen model [17].](image)

According to the model, force generated by the MR damper is given by

\[
F_d = C_2 \dot{y} + K_2 (x_1 - x_0) \tag{4}
\]

where \( \dot{y} = \frac{1}{c_1 + c_2} [\alpha z + c_1 \dot{x}_1 + K_1 (x_1 - y)] \)

\[
\dot{z} = -y|\dot{x}_1 - \dot{y}|z|z|^{n-1} - \beta(\dot{x}_1 - \dot{y})|z|^n + A_1 (\dot{x}_1 - \dot{y})
\]

\[
C_1 = C_{1a} + C_{1b} A, \quad C_2 = C_{2a} + C_{2b} A, \quad \text{and} \quad \alpha = \alpha_a + \alpha_b A
\]

\( n \) is a parameter representing the smoothness of transition from elastic to plastic response and in this paper it is chosen to be 2. \( z \) is hysteretic displacement. \( x_1 \) and \( y \) are MR damper piston displacement and internal displacement respectively. \( A \) is an input current applied. \( \beta, \gamma, \) and \( A_1 \) are parameters representing the control of linearity during unloading and the smoothness of the transition from the pre-yield to post-yield region. \( \alpha \) is a parameter representing stiffness for the damping force component associated with the evolution variable \( z \). \( K_1 \) is a parameter representing the control of the stiffness of the spring at higher velocities. \( K_2 \) is a parameter representing stiffness of the spring associated with the normal damper due to the accumulator.
C is a parameter representing viscous damping observed at higher velocities. \( C \) is a parameter representing the dashpot included in the model to produce the roll off at low velocities. \( x_0 \) is a parameter representing the initial displacement of the spring with the stiffness \( K \). Based on this model, control parameters that govern the hysteretic force and displacement of the damper have been identified. The shape of the hysteretic loop is controlled by the parameters \( \alpha, \beta \) and \( \gamma; A_1 \) is the restoring amplitude. The parameters \( C_1, C_2 \) and \( \alpha \) vary with respect to the applied input current. The hysteretic force increases with the increase in the input current. There are thus twelve parameters \( \alpha_a, \alpha_b, C_{1a}, C_{1b}, C_{2a}, C_{2b}, K_1, K_2, \gamma, \beta, A_1 \) and \( x_0 \) which decide the hysteretic force and its behavior in the modified Bouc-Wen model.

3 SOLUTION METHOD

Based on Modified Bouc-Wen Model, the twelve identified control parameters can be defined as a vector \( x = [\alpha_a, \alpha_b, C_{1a}, C_{1b}, C_{2a}, C_{2b}, K_1, K_2, \gamma, \beta, A_1, x_0] \). The experimental data for normal hip, thigh and shank motions taken from [5], are shown in Figure 4. For the given input experimental data, equation 3 has a unique solution of shank angle trajectory and hence prosthetic knee angle trajectory is defined. Therefore, the control parameters of the MR damper should be selected such that the knee angle trajectory for the swing phase should match the experimental knee angle trajectory and this is achieved by feeding an appropriate input current to the damper. This involves formulating an optimization problem, which will minimize the error between expected shank angle (\( \theta_se \)) from experimental data and computed shank angle (\( \theta_sc \)) from the dynamic equation of motion. Moreover, taking into account the effects of possible variations in the hysteretic characteristics of the MR damper, the lower and upper values of control parameters are selected and used as constraints. Variation in control parameters affects the hysteretic characteristics of the MR damper and hence the swing phase trajectory of shank. Knee swing phase trajectory is observed to be more sensitive to variation in stiffness and damping coefficient control parameters, \( C_{2a}, C_{2b}, K_1, \alpha_a, \alpha_b, C_{1a}, C_{1b} \) and \( K_2 \), than other parameters. By keeping the other control parameters fixed at certain local optimum values and varying any one of the control parameters such that the computed knee angle curve better approximates the expected knee angle curve, it is observed that an increase in \( C_{2a}, C_{2b} \) and \( K_1 \) increases the knee flexion angle and hence lengthens swing phase duration, whereas an increase in \( \alpha_a, \alpha_b, C_{1a}, C_{1b} \) and \( K_2 \) reduces the knee flexion angle and hence shortens swing phase duration. Variations in \( \gamma, \beta \) and \( A_1 \) are found to have less effect. Moreover, boundary limits of \( x_0 \) are chosen by taking into consideration the possible stroke length of the damper and the geometry of the swing phase leg model assumed. Within that limit variation, \( x_0 \) is also found to have less effect. Taking into account such observations, the lower and upper bound values of the control parameters are chosen and given in Table 1.

![Table 1. Control parameter bound values for the modified Bouc-Wen model](image)

Hence, the optimization problem may be defined as:

\[
\text{Min}_x(R(x) = \int_{t_0}^{t_f} (\theta_{se}(t) - \theta_{sc}(x,t))^2 \, dt)
\]

Subjected to: \( \text{Lower Bounds} \leq x \leq \text{Upper Bounds} \)

where \( t_0 \) and \( t_f \) are the start and end times of the swing phase.
Numerical algorithms for constrained nonlinear optimization are broadly categorized into gradient-based methods and direct search methods. Gradient-based methods use first derivatives or second derivatives. On the other hand, direct search methods of numerical algorithms for constrained nonlinear optimization problem such as differential evolution do not use derivative information and are more tolerant to the presence of noise in the objective function and constraint. Differential evolution is a simple stochastic function global minimizer which is also computationally expensive, but is relatively robust and works well for such kind of coupled system of equations. Hence, in this work, it is used to optimize the control parameters in Mathematica™ software using default set of values for the same.

Most of the commercially available MR dampers which are also suitable for such prosthetic knee application, such as LORD Corporation products: RD-1005-3, RD-8040-1 and RD-8041-1 operate within 0A – 1.5A. The values of the control parameters are optimized at 1.5A input current, which would be the maximum operational current in active mode. Then after numerically determining the optimal control parameters, the angular velocity of shank is compared with that of the normal person, such that it is within acceptable range to be stopped by an extension bumper to bring the knee angle to zero around the end of swing phase. Subsequently using optimal control parameters, the performance of the controller is further simulated to see how the controller behaves at 0A, when the damper is acting like a passive damper, and at 1A, when the battery power goes down.

4 RESULTS

Experimental data needed for the computation, adopted from [5], are taken for the normal walking of a person with 56.7 kg body mass at an average velocity of 1.3 m/s from toe-off to heel-strike and are shown in Figure 4. Physical parameters of the model are computed based on anthropometric table [5] for the person and listed as: M=56.7 kg, l₁=0.314 m, l₂=0.425 m, m₁=5.67 kg, a₁=0.136 m, m₂=3.46 kg, a₂=0.2576 m, I₁=0.058 kg m², I₂=0.108 kg m².

Attachment of MR damper to thigh as an offset from knee axis is taken to be equivalent to a maximum offset of gastrocnemius muscle attachment location on femur at lateral epicondyle and that of hamstring muscle on tibia at lateral condyle from knee axis [5], hence s is taken to be 0.05 m. Considering LORD RD-8040-1 as a suitable MR damper for such application which has 0.208 m length at fully extended position and 0.153 m length at compressed position with 0.055 m of stroke, and at a fully extended position of shank during the end of swing phase where knee angle is near zero, b is fixed to be 0.202 m. Considering the same damper, one can compute excitation displacement of the damper at toe-off, x₁(t), to be:

\[ x₁(t) = \sqrt{a^2 + b^2 - 2ab\sin(θ₀)} - l_s, \]  \hspace{1cm} (6)

where \( l_s \) is the compressed length of the MR damper.

The constrained optimization problem formulated in equation 5 has been optimized using differential evolution algorithm in Mathematica™ software. Obtained optimal control parameters are given in Table 2 and hence computed knee angle and expected knee angle are also shown in Figure 5.

For 1.5A input current, at which the optimal parameters are determined, the objective function is found to be 0.0035 rad², maximum knee flexion angle is 52.23°, and duration of swing phase is 0.341 s where the knee angle is 0°. The velocity of shank near the end of swing phase, at 0.3 s, is found to be 4.67 rad/s, which is small enough to be stopped by a rubber bumper. At about the same time, the normal person shank velocity [5] is 3.85 rad/s. Thus, with optimal control parameter and 1.5 A input current, when a normal motion for thigh is defined, single-axis knee with MR damper is able to produce proper swing phase motion of an amputee.
Table 2. Optimal values of control parameters

<table>
<thead>
<tr>
<th>Control Parameters</th>
<th>Optimal Values</th>
<th>Control Parameters</th>
<th>Optimal Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A_1$</td>
<td>1.953</td>
<td>$K_3$ (N/m)</td>
<td>1.359</td>
</tr>
<tr>
<td>$C_{iA}$ (Ns/m)</td>
<td>0.123</td>
<td>$\alpha_i$ (N/m)</td>
<td>4.158 x10^{-6}</td>
</tr>
<tr>
<td>$C_{iB}$ (Ns/m)</td>
<td>0.076</td>
<td>$\alpha_b$ (N/m.A)</td>
<td>0.653</td>
</tr>
<tr>
<td>$C_{2A}$ (Ns/m)</td>
<td>6.573 x10^{-6}</td>
<td>$\beta$ (1/m^2)</td>
<td>4.378</td>
</tr>
<tr>
<td>$C_{2B}$ (Ns/m.A)</td>
<td>2.742</td>
<td>$\gamma$ (1/m^2)</td>
<td>2.909</td>
</tr>
<tr>
<td>$K_1$ (N/m)</td>
<td>23.558</td>
<td>$x_0$ (m)</td>
<td>0.0005</td>
</tr>
</tbody>
</table>

Figure 4. Input experimental data from [5].

In such kind of powered prosthetic knee, systems work properly as long as the rechargeable battery produces appropriate current in the whole walking cycle. But this may not always be the case as the power may decrease throughout the operation. Therefore, the designed controller must be checked for robustness in such cases. For this purpose, using designed optimal controller parameters, the system was simulated at 0 A where the power is in off-state and the knee is working like a passive single-axis knee and at 1 A where the power is reduced to some extent in the processes of walking. The results are displayed in Figure 5. At both 1 A and 0 A current, the knee flexion angle is within the acceptable range of the normal person, which is 0° to 60.56°. The duration of swing phase at 1 A current is nearly the same as that of the normal person, whereas at 0 A current an amputee may experience a shorter swing phase duration. This is because the damping coefficient elements of the damper are reduced and hence the damper is less viscous in passive mode. The simulation results are summarized in Table 3.
5 CONCLUSIONS

In this work, a controller for single axis prosthetic knee has been designed for trans-femoral amputees. The controller is an MR damper, twelve control parameters of which are determined through minimization of error between the expected knee angle data with the computed knee angle from formulated dynamic system when thigh motion data is feed as input to the system. By studying variation effects of each parameter, the upper and lower boundary values are fixed and optimization problem is solved as a constrained optimization problem by global optimization technique to determine a set of control parameters. Moreover, with these optimal parameters, conditions of knee angle and shank velocity at the end of swing phase have been checked. The robustness of the controller is also checked for two possible extreme conditions that an amputee with such controller might experience. Overall, a platform of designing MR damper for semi-active type above-knee prostheses has been given. This work should be extended with experimental data of walking on rough terrain, step climbing, jumping and the like. Other prosthetic knee models such as four bar polycentric knee can be used to increase its performance.

REFERENCES


Development of a human-robot dynamic model to support model-based control design of an upper limb rehabilitation robot

B. Ghannadi*, N. Mehrabi*, J. McPhee*

* Systems Design Engineering
University of Waterloo
200 University Ave. West, N2L3G1 Waterloo, ON, Canada
{bghannad, nmehrabi, mcphee}@uwaterloo.ca

ABSTRACT
Rehabilitation robots are developed to treat the preponderance of movement disorders caused by stroke, brain and spinal cord injuries. These robots can facilitate repetitive and intense therapy sessions. One of the neglected aspects in the design and control of rehabilitation devices is the dynamic modeling of the human-robot interaction. In this study, for the model-based control design of an upper limb rehabilitation robot, a two dimensional (2D) human-robot dynamic model is developed. To this end, a method to reduce a three dimensional (3D) musculoskeletal arm model to a 2D model is implemented. The performance of the 2D upper limb model is assessed by simulating the 3D upper extremity musculoskeletal model interacting with the robot. Results show that it is possible to use the 2D musculoskeletal arm model to evaluate the controller of a planar rehabilitation robot.

Keywords: Biomechatronics, Upper extremity, Rehabilitation robot, Human-robot model, Musculoskeletal model reduction.

1 INTRODUCTION
There are a large number of people with movement disorders who have difficulties doing their daily tasks independently [1]. Much research has been devoted to the therapy of these people, and rehabilitation robots have been developed to assist with therapy [2, 3, 4, 5, 6]. In robotic rehabilitation, since the human body is interacting with a mechanical device, safety issues in the design of appropriate control strategies are very important [7]. Thus, rehabilitation robots usually use control approaches that consider the human body interacting with the robot as a mechanical impedance [8]. Characteristics of this mechanical impedance may vary depending on different musculoskeletal factors such as posture and muscle contraction dynamics [9], and multiple experiments are required to evaluate them. However, in the current state of the art, rehabilitation robot controllers conservatively assume the robot is interacting with a static impedance model. In other words, there is a lack of research that considers realistic human body dynamic interactions with the rehabilitation robot.

Since human-robot interaction affects therapy procedures, the objective of the current study is to develop an integrated human-robot dynamic model with real-time simulation capability to support model-based control design. Upper limb motor defects are common among stroke patients [10], so this study is focused on an upper extremity rehabilitation system. In this research, an upper-extremity musculoskeletal model of a human arm is developed to simulate the patient’s arm movements. This arm model interacts with a model of an upper limb rehabilitation robot, which was developed by Quanser Inc. and the Toronto Rehabilitation Institute (TRI) [11].

This paper is organized as follows. First, in the Method section, the integrated human-robot model, planar and 3D musculoskeletal arm models, and simulation description are provided. Next, in the Results and Discussion section, results of human-robot system simulations are discussed. Finally, in the Conclusion section, potential contributions and future work are presented.
2 METHOD

In this section, first, the integrated human-robot model is provided. Then, development of different musculoskeletal arm models are discussed. Finally, the procedure to simulate human-robot interactions is described.

2.1 Integrated human-robot model

The studied rehabilitation robot is a planar parallelogram arm with 2 degrees of freedom (DOF); to achieve simplicity and low computational cost, a 2D musculoskeletal arm model is used to interact with the robot [12] (see Figure 1). Because of its multi-domain capabilities, symbolic processing, and optimized code generation, the MapleSim software is utilized to model and simulate the integrated human-robot system in this study.

![Figure 1](Image)

2.2 Musculoskeletal arm model

2.2.1 Primary 2D arm model

The 2D arm model, which is common for studying reaching movements in the horizontal plane [13, 14, 15, 16], has two hinged links with 6 muscle groups including shoulder mono-articular flexor (Muscle 1), shoulder mono-articular extensor (Muscle 2), elbow mono-articular flexor (Muscle 3), elbow mono-articular extensor (Muscle 4), shoulder-elbow bi-articular flexor (Muscle 5), and shoulder-elbow bi-articular extensor (Muscle 6). Physical parameters of the links and muscle insertion points are adopted from [14]. It is assumed that there is no tendon compliance in the musculotendon structure\(^1\), and only the contractile element of the Hill-type muscle [18] is generating muscle force (Equation (1)).

\[
F_M = a_M f(F_{iso}, L^M, V^M)
\]  

(1)

where \(F_M\), \(a_M\), \(L^M\) and \(V^M\) are the \(M\)th muscle group force, activation, length and velocity, respectively, and \(f(.)\) is the contractile element function derived from [18]. Muscle model parameters are summarized in Table 1.

The muscle force sharing problem is solved using the forward static optimization (FSO) approach.

---

\(^1\)The compliance of tendon is proportional to its slack length. Thus, a tendon is compliant if its normalized slack length is large (\(\geq 10\)), and it is very stiff when it is equal to 1 [17]. For most muscles in the upper extremity this value is around 1; hence, the stiff tendon assumption seems to be valid.
Table 1. Parameters for the primary 2D arm muscle model, where $F_{iso}$ is the maximum isometric muscle force, $L_{0}^{M}$ is the optimal muscle length, $L_{s}^{t}$ is the tendon slack length, and $\alpha_{0}$ is the muscle pennation angle at its optimal length.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Muscle 1</th>
<th>Muscle 2</th>
<th>Muscle 3</th>
<th>Muscle 4</th>
<th>Muscle 5</th>
<th>Muscle 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>$F_{iso}$ (N)</td>
<td>1000</td>
<td>1000</td>
<td>1000</td>
<td>1000</td>
<td>1000</td>
<td>1000</td>
</tr>
<tr>
<td>$L_{0}^{M}$ (cm)</td>
<td>12.5</td>
<td>5.6</td>
<td>14.2</td>
<td>10.1</td>
<td>37</td>
<td>24.9</td>
</tr>
<tr>
<td>$L_{s}^{t}$ (cm)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$\alpha_{0}$ (deg)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Viscous damping coefficients are assigned to the shoulder and elbow joints. Following [20], their values are $1.3$ (Nms/rad) for the shoulder joint, and $0.3$ (Nms/rad) for the elbow joint. The cost function for the FSO is:

$$J_{FSO} = K_{1} \sum_{i=M}^{6} \alpha_{M}^{2} + K_{2} \left[ (K_{X}\Delta X)^{2} + (K_{Z}\Delta Z)^{2} + (K_{X}\Delta \dot{X})^{2} + (K_{Z}\Delta \dot{Z})^{2} \right]$$

where $\Delta X$ and $\Delta Z$ are the position errors in tracking the desired path, $\Delta \dot{X}$ and $\Delta \dot{Z}$ are the velocity errors in $X$ and $Z$ directions, and $K_{c}$’s are the weighting constants which are tuned manually.

2.2.2 3D musculoskeletal arm model

To validate the 2D arm model, a static optimization (SO) followed by an inverse kinematics and dynamics analyses is performed on an equivalent 3D musculoskeletal arm model. This model is based on [21], which is available in the OpenSim open-source software package [22]. This 3D model has 4 DOF (3 DOF at shoulder and 1 DOF at elbow) and is developed in MapleSim (Figure 2a). It includes 22 muscles (12 muscle units) of the shoulder, elbow and forearm with the same muscle model as the 2D arm. These 22 muscles are listed in [13].

In the 3D model, the shoulder joint is modeled by three revolute joints with intersecting axes. These body-fixed rotation axes are as follows ($j$ is the joint/axis number):

- $j = 1$: Shoulder plane of elevation (in OpenSim: "elv_angle") axis (Y axis); rotation about this axis deviates the upper arm from the thorax in the transverse plane.
- $j = 2$: Shoulder elevation angle (in OpenSim: "shulder_elv") axis (rotated body-fixed Z axis); rotation about this axis results in upper arm abduction/adduction.
- $j = 3$: Shoulder rotation (in OpenSim: "shulder_rot") axis (rotated body-fixed Y axis); rotation about this axis causes internal/external rotation of the upper arm. The shoulder rotation angle is measured with respect to the shoulder plane of elevation; hence, actual shoulder rotation angle is the total amount of rotation about this axis minus the shoulder plane of elevation angle.

The elbow joint ($j = 4$) is a revolute joint defined by "elbow.flexion" axis in OpenSim. To generate a 2D-equivalent of the 3D musculoskeletal arm model the following steps are accomplished:

1. Scaling the 3D model based on the ratio of its upper arm length to the 2D model first link (upper arm) length.
2. Finding proper YZY rotation (shoulder rotation), so that the elbow flexion-extension axis becomes perpendicular to the transverse plane.
3. Assigning the center of mass and mass moments of inertia\(^2\) of the segments such that, by following the mentioned YZY rotation, their values become identical with the 2D model inertial properties.

Musculotendon lengths and moment arms are evaluated after the inverse kinematics analysis of the 3D model in OpenSim. Finally, after inverse dynamics analysis, the muscle redundancy problem is solved by SO with the following cost function:

\[
J_{SO} = K_1 \sum_{m=1}^{22} a_m^2 \tag{3}
\]

subject to:

\[
T^j = \sum_{m=1}^{22} r_m^j F_m^j \tag{4}
\]

where \(m\) and \(j\) denote the muscle and joint numbers, respectively. \(T\) is the evaluated joint torque, \(r, a\) and \(F\) are the muscle moment arm, activation and force, respectively.

![Figure 2](image.png)

**Figure 2.** (a) The 3D musculoskeletal arm model (adapted from the OpenSim software upper extremity model). (b) The modified 2D musculoskeletal arm model.

### 2.2.3 Modified 2D arm model

In the simulation results, we noticed that the activation results of the 3D model simulation are different from the 2D model ones. Therefore, we modified the 2D model using a weighted average approach to lump the 22 muscles of the 3D model into 6 muscle groups in the 2D model (see Figure 2b). In the current work, we have modified the weighted average method used in [13]. The procedure of this modified model reduction is as follows:

1. Define muscle groups based on their synergistic muscles:

   - **Muscle 1:** Anterior and Middle Deltoid; and Clavicular, Sternal and Ribs Pectoralis Major.
   - **Muscle 2:** Posterior Deltoid; Thoracic, Lumbar and Iliac Latissimus Dorsi; and Teres Major.
   - **Muscle 3:** Brachioradialis; Brachialis; Pronator Teres; and Extensor Carpi Radialis Longus.

\(^2\)Based on anthropometric data [23], ellipsoids (with three unknown radii and two anthropometric constraints related to their ratios) are fitted to the upper and lower arms so that after prescribed rotations, they possess the correct moments of inertia in the horizontal plane.
Muscle 4: Extensor Carpi Radialis Brevis and Ulnaris; Triceps Lateral and Medial; Anconeus.

Muscle 5: Biceps Long and Short.

Muscle 6: Triceps Long.

2. Calculate muscle moment arms about the three revolute shoulder joints and one revolute elbow joint for a specific type of motion derived from inverse kinematics analysis in OpenSim. Evaluate the shoulder moment arm in the horizontal plane ($r_{S_{m,M}}$).

\[
 r_{S_{m,M}} = \sum_{j=1}^{3} R^G_{j\{2,3\} \rightarrow (2,2)} r_j^j
\]

where $r_j^j$ is the muscle moment arm about the corresponding $j$th joint. Subscripts $m$ and $M$ denote muscle number and its corresponding muscle group number, respectively. $R^G_{j\{p,q\}}$ is the $(p,q)$th element of the rotation matrix converting the joint $j$ local frame to the global frame.

3. Evaluate the corresponding maximum shoulder and elbow rotation torques for individual muscles in the horizontal plane.

\[
 \begin{align*}
 T_{S_{m,M}} &= r_{S_{m,M}}^m T_{S_{m,M}}^\text{iso} \\
 T_{E_{m,M}} &= r_{E_{m,M}}^m T_{E_{m,M}}^\text{iso}
\end{align*}
\]

where $r_{E_{m,M}}^m = r_j^j$ is the elbow moment arm in the horizontal plane.

4. Calculate the corresponding maximum shoulder and elbow torques for the muscle groups in the horizontal plane (Equation (7)). Then, evaluate the muscle group moment arms (Equation (8)).

\[
 \begin{align*}
 T_{S_{m,M}} &= \sum_{m=1}^{22} T_{S_{m,M}}^m \\
 T_{E_{m,M}} &= \sum_{m=1}^{22} T_{E_{m,M}}^m \\
 r_{S_{m,M}} &= \sum_{m=1}^{22} r_{S_{m,M}}^m T_{S_{m,M}}^m \\
 r_{E_{m,M}} &= \sum_{m=1}^{22} r_{E_{m,M}}^m T_{E_{m,M}}^m
\end{align*}
\]

5. Approximate the muscle group insertion points by running an optimization on origin and insertion locations of the 3D model in a way that a straight line path model generates similar moment arms over a period of inverse kinematics motion.

6. Evaluate the muscle group model parameters ($\zeta$).

\[
 \zeta_M = \frac{1}{2} \left( \frac{\frac{1}{t} \sum_{m=1}^{22} \xi_M T_{S_{m,M}} T_{S_{m,M}}}{\frac{1}{t} \sum_{m=1}^{22} T_{S_{m,M}}} + \frac{\frac{1}{t} \sum_{m=1}^{22} \xi_M T_{E_{m,M}} T_{E_{m,M}}}{\frac{1}{t} \sum_{m=1}^{22} T_{E_{m,M}}} \right)
\]

where $t$ is the simulation time, and $\zeta_M$ denotes the parameters listed in Table 2.

7. Since we are not including muscle wrapping, negative values of the tendon slack lengths of the muscle groups are set to zero so that the value of the muscle group musculotendon length becomes meaningful.

Finally, muscle attachment points are updated as in Figure 2b and the new muscle model parameters are summarized in Table 2.
Table 2. Parameters for the modified 2D arm muscle model.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Muscle 1</th>
<th>Muscle 2</th>
<th>Muscle 3</th>
<th>Muscle 4</th>
<th>Muscle 5</th>
<th>Muscle 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>$F_{iso}$ (N)</td>
<td>2525</td>
<td>1672</td>
<td>1452</td>
<td>1577</td>
<td>972</td>
<td>799</td>
</tr>
<tr>
<td>$L_0^M$ (cm)</td>
<td>12</td>
<td>21</td>
<td>11.7</td>
<td>10.9</td>
<td>12.9</td>
<td>14.3</td>
</tr>
<tr>
<td>$L_1^M$ (cm)</td>
<td>2.9</td>
<td>0</td>
<td>1.8</td>
<td>0.7</td>
<td>18.8</td>
<td>11.9</td>
</tr>
<tr>
<td>$\alpha_0$ (deg)</td>
<td>21.6</td>
<td>19.5</td>
<td>1.4</td>
<td>7.8</td>
<td>0</td>
<td>12</td>
</tr>
</tbody>
</table>

2.3 Simulations

The rehabilitation robot is designed to perform point to point reaching tasks in the horizontal plane [24]. In our simulations of reaching movements, the manipulation trajectory is approximated by a smooth circular path with a large radius of curvature (Figure 1). A cubic spline interpolation approach is used to generate this path with a bell-shaped tangential speed profile and continuous jerk. In the human-robot simulations, an unactuated robot is interacting with the aforementioned musculoskeletal arm models. Considering the limited range of joint motions of the OpenSim 3D musculoskeletal model (i.e., shoulder rotation angle because of imperfect muscle wrapping is 20 degrees, while this value in most references have been reported to be about 90 degrees [25]), the system is simulated only for about one fifth of the reaching path (i.e., 2 seconds).

Figure 3. Muscle activations of the three upper limb musculoskeletal models interacting with the unactuated robot.
3 RESULTS AND DISCUSSION

The results are summarized in Figure 3, in which we see that the muscle activations of the modified 2D model are a better match to the 3D model than the primary 2D model. The reasons for the differences between muscle activations of the 3D and modified 2D musculoskeletal arm models can be summarized as follows:

1. An approximate method is used for lumping muscles from the 3D model into the 2D model.
2. A straight-line approach for muscle paths in the 2D model will decrease active muscle lengths. This has a significant influence on the behavior of the shoulder muscles, which have complicated wrapping geometries (note the initial activation of Muscle 2).
3. The 3D model has three shoulder torque constraints (Equation (4) for $j = 1..3$), while the 2D model has only one shoulder torque constraint.

4 CONCLUSIONS

In the current work, we show that the 2D musculoskeletal arm model can effectively represent the 3D motion of the arm; thus, it can be used to evaluate the rehab robot performance. Although development of this 2D musculoskeletal arm model requires a 3D model with muscle wrapping geometries, the lower-fidelity 2D model—in contrast to the higher-fidelity 3D model—can be used in real-time simulations and model-based controllers. Currently, a subject-specific 3D musculoskeletal arm model with proper muscle wrapping based on the obstacle-set method is being developed. By the presented model reduction method, the 2D musculoskeletal model with muscle wrappings can be developed to represent different patients. For validation of the model and controller, healthy subjects will be recruited to perform predefined reaching movements with the robot (working with a model-based controller); the end-effector force, robot joint angles, surface electromyography and kinematics of the upper limb will be monitored.

ACKNOWLEDGEMENT

This work was funded by the Natural Sciences and Engineering Research Council of Canada (NSERC). The authors thank Quanser Inc. for providing the upper limb rehabilitation robot, and the Toronto Rehabilitation Institute (TRI) for many fruitful discussions.

REFERENCES


Development of a Musculotendon Model within the framework of Multibody Systems Dynamics

Ana R. Oliveira*, Sérgio B. Gonçalves*, Mamede de Carvalho#, Miguel T. Silva*

*IDMEC, Instituto Superior Técnico
Universidade de Lisboa
Lisbon, Portugal
[Ana.r.sousa.de.oliveira, Sergio.goncalves, MiguelSilva]@tecnico.ulisboa.pt

# Faculty of Medicine
Universidade de Lisboa
Lisbon, Portugal
mamedealves@medicina.ulisboa.pt

ABSTRACT

Despite appearing a simple task, human movement is the result of a complex and synergistic interaction between the musculoskeletal and central nervous system. As a result of a composite pattern of neural excitations, muscles contract coordinately to produce forces that are transmitted by tendons to the skeletal system, causing its movement or keeping its pose. Often undervalued in the current muscle models, the elastic properties of tendon present an important action in the dynamic interaction between the muscular and skeletal system, influencing the force transmission, energy storage and joint control.

This work aims to develop a musculotendon model within the framework of multibody systems dynamics with natural coordinates to be applied both in inverse and forward dynamics analyses of human body. An efficient methodology to compute the musculotendon forces and activations is presented, considering the Hill-type muscle model as basis. The proposed model considers a muscle unity in series with a spring element, which is defined according the elastic properties of the tendon.

The assessment of the methodology is performed by applying the proposed model in the inverse dynamic analysis of a 3D human biomechanical model, characterized by 43 muscles per leg. The influence of the tendon in the produced muscle force and activation is analysed for different daily activities and the results compared to a biomechanical model with stiff (rigid) tendons.

The results indicate that the tendon allows the muscle to work, predominantly, in its optimal configuration, decreasing the non-physiological passive forces and the activations needed to perform the required contractile force.

Keywords: Musculotendon Model, Tendon Model, Muscle Model, Musculotendon Contraction Dynamics, Multibody Dynamics

1 INTRODUCTION

Last decades were marked by a great number of works exploring the human movement, in part, due to their applicability in different areas such as biomechanics, health and sports science, prosthetics and orthotics, robotics, among others. Scientific research in these fields is of particular interest, since it allows for a better understanding of normal and abnormal human movement characteristics, as well as the development of new and innovative ways to increase human performance and quality of people’s life.

Despite being performed since birth, human movement is the result of an intricate process involving the musculoskeletal and the central nervous system (CNS). A complex pattern of electric signals, generated at CNS level, results in a synergetic contraction of muscle fibers, which induces forces that are transmitted by tendons to the skeletal system, causing its movement or maintaining a given pose. As such, the study of muscle activity is of great interest for the scientific and medical community alike, as it allows for better understanding the contribution of a specific
muscle to a given movement [1], providing insight to the design of fully-customized prostheses, orthoses, functional neuromuscular stimulation systems and other assistive devices [2], among other applications.

An important aspect of the musculotendon complex arises from the viscoelastic properties of the tendon, which allow for a dynamic interaction between the muscular and skeletal system, influencing the force transmission, energy storage, joint control and movement accuracy [3]. Hence, the development of non-invasive methods based on musculoskeletal modelling and computer simulations is of particular interest, since these methodologies allow to understand the dynamics underlying these interactions and their influence on movement.

Several mathematical models have been presented to depict the contraction dynamics of muscles, standing out the ones proposed by Huxley [4] and Hill [5]. Based on the physiological structure of muscles, Huxley-type models estimate the muscles forces considering the cross-bridges between the actin and myosin filaments. Despite presenting a strong physiological component, each muscle in this model is ruled by a set of differential equations, one per cross-bridge, increasing substantially the computational complexity required to solve the problem [6]. On the other hand, Hill-type models focus more on the mechanical behaviour of the muscle, considering, in most of cases, only one differential equation per muscle. As result, the computational effort is not as high as in the Huxley model, making this type of models suitable for large-scale muscle simulations [6]. Regardless of the large amount of works applying computational muscle models, few discuss the tendon modelling. Almost all methodologies found in literature implement or adapt the model proposed by Zajac [2,7], in which the viscoelastic properties of the tendon are described by a spring element with constant stiffness.

The main objective of this work is the development of a musculotendon model within the framework of multibody systems dynamics that takes into account the influence of tendon in muscle contraction. The methodology applied in the solution of the problem should be robust enough, enabling its application both in forward and inverse dynamics simulation of large biomechanical systems. The proposed model is based on the work developed by Zajac [2], in which the musculotendon unit is described by a Hill-type muscle model in series with an elastic element that depicts the physiological properties of the tendon.

The methodology required to compute the variables of the model was embedded in the simulator APOLLO [3], a built-in three-dimensional multibody dynamics software with natural coordinates. The assessment of the model was performed by applying it in the study of three daily activities with different levels of tendon recruitment: walking, running and jumping. For this purpose a human biomechanical model (lower limbs and HAT), with 43 musculotendon unities per leg, was applied. The influence of the tendon in the muscle contraction dynamics is analysed by comparing the obtained results (activations, muscle/tendon forces and lengths) with a musculotendon model in which the tendon is modelled assuming stiff properties, i.e. the length of the tendon is considered constant and equal to the respective slack length [8].

2 MUSCULOTENDON MODEL

The physiological musculotendon dynamics begins with the generation of a neural activation signal, which excites a given muscle unity leading to its contraction. In order to model this complex dynamic, it is important to understand the mechanical behaviour of the muscle tissue subjected to different neural signals. As such, the model should contemplate an activation dynamics, which describes the transformation of a neural excitation $u(t)$ in the respective muscle activation $a(t)$, and a contraction dynamics that correspond to the conversion of the muscle activation $a(t)$ in musculotendon force ($F_{MT}$) [1,6]. The Hill-type model, presented in Figure 1, is used in this work to represent muscle contraction dynamics, since it considers in its description both the mechanical properties of the muscle and tendon. An Hill-type model is generically described as a passive element, representative of the tendon, assembled in series with a muscle element angled by a pennation angle to model oblique insertions ($\alpha$).
Figure 1. Schematic representation of the Hill-type model with elastic tendon.

As such, the musculotendon length, represented by $l^{MT}$, is defined as the sum of the tendon length ($l^T$) and the muscle fibers length ($l^M$) taking into account $\alpha$ as represented in Eq. (1).

$$l^{MT} = l^T + l^M \cos(\alpha)$$

(1)

The tendon element is defined as a spring-like component with a constant stiffness $K_t$ that depends on the physiological elastic properties of the tendon (see Fig. 2.c). In turn, the muscle is represented by a contractile element (CE) assembled in parallel with a passive element (PE). The CE produces a force that depends on the force-length ($f_l(l^M)$) and force-velocity ($f_v(v^M)$) relationships (see Fig. 2.a and 2.b) proper of the muscle tissue and on its activation level ($a^M$). On the other hand the PE, which is used to simulate the elastic properties of the muscle, generates a force that depends only on the muscle length [3]. Hence, the resultant muscle force can be defined as the sum of the forces generated in these two components:

$$F^M = F^M_{CE} + F^M_{PE} = \frac{f_l(l^M)f_v(v^M)}{F^M_0}a^M + F^M_{PE}(l^M)$$

(2)

where $F^M$ is the muscle force, $F^M_{CE}$ and $F^M_{PE}$ represent respectively the contractile and passive force, and $F^M_0$ is the peak isometric active force.

Figure 2. a) Active and Passive Muscle Force-length relation. b). Force-velocity Muscle Curve. c) Force-Strain Tendon Curve.

As the tendon is placed in series with the muscle component, the musculotendon force is equal to the force transmitted by the tendon, which is in turn equal to a function of the muscle force and the respective pennation angle:

$$F^{MT} = F^T = F^M \cos(\alpha)$$

(3)
The contraction dynamics of the musculotendon is characterized by a first order differential equation, which relates the variation of the normalized musculotendon force for a fully-activated state \( \bar{F}_a^T \) with the contraction velocity of the tendon element and its stiffness [2,9]:

\[
\frac{\partial \bar{F}_a^T}{\partial t} = \bar{K}_t v^T
\]

where \( \bar{K}_t \) is calculated according to [2] \( \bar{K}_t = 30 \frac{\text{l}_{\text{L}-T}}{\text{l}_{\text{opt}}} \) and \( v^T \) is the tendon velocity, which can be defined considering the musculotendon velocity \( v^{MT} \) and muscle velocity \( v^M \) as described below:

\[
v^T = v^{MT} - \frac{v^M}{\cos(\alpha)}
\]

This way, the estimation of the musculotendon force requires the computation of \( v^T \), and consequently, \( v^M \), \( v^{MT} \) and \( \alpha \). Figure 3 describes the steps needed to calculate all this variables. Essentially, for each iteration, the model computes \( \bar{F}_a^T \) through numerical integration, considering as input the musculotendon force calculated in the previous time step, and \( l^{MT} \) and \( v^{MT} \), which can be directly determined from the system kinematics.

In order to enable the calculation of the muscle velocity and pennation angle, the tendon length and the contractile and passive muscle forces must be firstly determined. The first step comprehends the determination of the tendon length taking into account its own elastic properties (Fig. 3). The normalization of tendon slack length by the optimal fiber length, denoted by \( l^{\text{opt}}_T \), defines the compliance of the tendon. A stiff tendon \( (l^{\text{opt}}_T \leq 1) \) will be treated as inextensible, which implies that its length does not change over time, i.e., it is equal to the slack length and as so the tendon velocity is zero. On the other hand, a compliant tendon will be described by the inverse force-strain curve (Fig. 2.c) that could be represented as follow [2,9]:

\[
l^T(\bar{F}_a^T) = \begin{cases} 
I^T (1 + \frac{\ln \left( \frac{\bar{F}_a^T + 1}{10.10377} \right)}{91}), & 0 \leq \bar{F}_a^{MT} \leq 0.3086 \\
I^T (1 + \frac{\bar{F}_a^T + 0.26029}{37.526}), & 0.3086 \leq \bar{F}_a^T
\end{cases}
\]

The second step comprises the estimation of the pennation angle by assuming a constant muscle thickness \( l^W \).

\[
l^W = l^M \sin(\alpha_0) = l^T \sin(\alpha)
\]
The algebraic manipulation of the Hill-type model in Eq. (2) and (3) enables to define an expression in order to \( v^M \) as a function of the muscle maximum contractile velocity \( v_0^M \) and of the force-velocity relation:

\[
v^M = v_0 f^{-1}(f/v)
\]

(8)

So the force-velocity relation is given by:

\[
fv(v^M) = \frac{\hat{F}^T - F_0^M}{f(I^M) \cos(\alpha)} = \frac{F^M_{CE}(I^M)}{f(I^M)}
\]

(9)

The analysis of Equation (9) reveals two singularities, when \( a^M \to 0 \) and \( f(I^M) \to 0 \). To ensure that the \( fv \) presents a solution with physiological value, the muscle activation is assumed to be
equal to 1 \((a^M = 1)\) and the muscle force-length relationship higher than 10% of the muscle isometric active peak force \( (f(l^M) > 0.1F_0^M \to 0) \) [7].

Finally the last step considers the integration of this methodology in a multibody dynamics framework. The musculotendon forces computed using the methodology described above are applied as external forces of the model, considering a transformation to the equivalent generalized forces expressed in terms of the natural coordinates of the rigid bodies \(g\) [3,10]:

\[
M\ddot{q} + \Phi^T \lambda = g^{ext} + \sum_{i=1}^{\text{num}} g_{i1}^{MT} a^M_i \cos(\alpha_i) + \sum_{i=1}^{\text{num}} g_{i2}^{MT} \cos(\alpha_i)
\]  

(10)

where \(M, \ddot{q}\) and \(\Phi\) are respectively the mass matrix, generalized accelerations vector and Jacobian matrix of the biomechanical system. Vectors \(\dot{g}\) and \(\lambda\) present respectively the generalized force vector for a fully-activated muscle state and the Lagrange multipliers.

The activations and the muscle forces can be estimated by means of optimization methodologies:

Find: \(\bar{\lambda} = \{\lambda, a\}\)

Minimize: \(\mathcal{J}(\bar{\lambda})\)  

Subject to:

\[
\begin{align*}
\mathbf{f}_{eq} &= 0 \\
\lambda_i^{lb} &< \lambda_i < \lambda_i^{ub} \\
0 &\leq a^M \leq 1
\end{align*}
\]

(11)

where \(\mathcal{J}(\bar{\lambda})\) is the objective function defined according physiological criteria (e.g. minimization of the metabolic energy, muscle stress, muscle fatigue, among others) that is to be minimized respecting the equations of motion \(\mathbf{f}_{eq}\) for the defined biomechanical model and the bounds of the state and control variables.

3 RESULTS AND DISCUSSION

The methodology described in the section 2 was embedded in the built-in software APOLLO, a three-dimensional multibody dynamics software with natural coordinates [3]. An important aspect of the proposed model is its robustness, which enables its application both in inverse and forward dynamics simulations.

The assessment of the model was performed by applying it in the study of the musculotendon dynamics for three different activities: walking, running and jumping. This choice had in consideration the different expected levels of tendon recruitment in these movements. A human articulated biomechanical model composed by 12 segments (three-segment foot, shank, thigh, pelvis and HAT) was implemented using natural coordinates. Its formulation considers 20 rigid bodies defined by 20 points and 29 unit vectors, allowing a total of 27 degrees-of-freedom. In order to depict with detail the muscle apparatus of the lower limbs, 86 muscles (43 per leg) were integrated, considering the muscle/tendon parameters presented in [11].

The acquisition of the experimental data for a male subject (23 years old, 76 kg, 1.78 m) with no history of gait disorders was performed in the Laboratório de Biomecânica de Lisboa (Lisbon Biomechanics Laboratory) at Instituto Superior Técnico. The acquisition of kinematic data made use of a three-dimensional motion capture system, composed by 14 infrared reflective cameras (Qualisys ProReflex 10 x MC1000 and 4 x MC500). Three force plates (AMTI OR 6-7-1000, 508mm x 464mm) were used to acquire the ground reaction forces (GRF). The markers kinematics and the GRF were acquired synchronously using the Qualisys Tack Manager 2.9 (Qualisys) software. The sampling frequency was defined as 100 Hz for the IR cameras and 1000Hz for the force plates. A set of 10 valid trials were acquired for each activity. The statistical treatment was performed using MATLAB software.
In order to study the influence of including the tendon in muscle models, the output of the analysis for the three activities is compared with the results obtained for a second model in which the tendons are considered stiff, i.e. constant tendon length (equal to slack length). Due to the large amount of data, the results presented in this work will focus only in the triceps surae group (plantar-flexors), composed by the gastrocnemius lateral and medial and the soleus. The results obtained for the other muscles can be consulted in [12].

3.1 Gait, Running and Jumping

In general, the patterns of force and activation are consistent with the performed movement, i.e. variations in the curve shape occur during the events in which the respective muscles are recruited. The analysis of the gait results (Fig. 5.a) allows the observation of a peak of force and activation during the mid-stance (~30-50% of gait cycle (GC)) and pre-swing (~50-60% GC) phases. It is known that the mid-stance is characterized by a storage of elastic energy in the muscles of the triceps surae [13], which can be identified by analysing the increase of musculotendon force in these muscles. The stored energy is posteriorly released during the push-off phase (40-60%). This phase is also characterized by a concentric contraction of these muscles in order to plantar flex the ankle and propel the body forward. Once more, this sequence of events is also observed in the force and activation curves for this phase. The results show an increase of force and activation values until approximately 40-50% of GC, followed by a decrease during the last moments of the impulsion phase. Moreover, similar curves are obtained when the activation patterns are compared with electromyographic data (EMG), attesting the good applicability of the methodology [13]. For instance, the analysis of the EMG signal for the triceps surae muscles shows also a peak between the 30 and 50% of GC, in line with the powered plantar flexion that occurs during propulsion. A consistent pattern of recruitment is also observed in the running and jumping movements (Fig. 5.b and 5.c). The phases characterized by energy storage and concentric contraction present also the expected variations in the musculotendon force and activation (running: 10-40%, jumping: 20-45%).

The comparative analysis of the two models addressed in this work allows to observe significant differences both in the pattern and magnitude of the curves. The model with stiff tendon presents non-physiological muscle forces caused by the passive component of the Hill-type model more frequently than its counterpart. On the other hand, the comparison between the forces produced by the contractile elements (Fig. 5.a) and the musculotendon forces for the compliant model shows small differences, indicating that the presence of passive forces are almost non-existent. Moreover, despite presenting lower values of contractile forces, the activations for the stiff model are higher. This fact is also expected, since as the muscle is working away from its optimal zone ($l_0^M$), it requires a high state of activation to achieve the same contractile force. For instance, the magnitudes of contractile forces obtained for the gastrocnemius medialis in the walking are respectively 500N for an activation of 0.4 (compliant model) and 135N for an activation of 1 (stiff model).
The results also show that the inclusion of a compliant tendon influences the contraction dynamics of the muscle by avoiding its excessive extension. This behaviour was already expected, since modelling the tendon as elastic element enables its own extension, storing energy and avoiding that all the length variations on the musculotendon element would be borne by the muscle segment. The results indicate that the influence of modelling the tendon as an elastic element is as important as the level of muscle recruitment and the forces involved (ex: running and jumping), improving the physiological significance of the analysis.

Lastly, there were no significant differences between the computation times for the two models, as well as, no problems were found regarding the convergence and the application of the methodology proposed in this work.
CONCLUSIONS

This work aimed to the development of a valid musculotendon model to be applied in the inverse and forward dynamic analysis of multibody systems with natural coordinates. The proposed methodology allowed to compute with success all the required variables to model the musculotendon dynamics. No problems were found with the convergence of the method, as well as, no significant differences were observed in the optimization time between the two models considered in this work (stiff and compliant).

The model outputs were consistent with the performed movement and the respective EMG pattern, attesting the applicability of the method. The comparison of the two models indicates a clear influence of the tendon in the musculotendon contraction dynamics for all the studied movements. In general, the inclusion of the tendon resulted in an adaptation of the muscle length, working nearer its optimal zone. Its inclusion avoided also non-physiological passive forces resultant of the excessive extension of the muscle component. In general, the results indicate that the inclusion of the tendon confers a more physiological significance to the simulation, pointing that its use should not be neglected even in the study of slower movements, such as walking.

REFERENCES


Gait and Posture responses to backpack load

Mohamed A. Sayari #, Amur Al Yahmedi*, and Neila Masmoudi #

*Mechanical & Industrial Engineering
College of Engineering
Sultan Qaboos University
P.O. Box 33, Al-Khod 123
Sultanate of Oman

# Electromechanical Systems Laboratory (LASEM),
National Engineering School of Sfax
University of Sfax, BP 1173
3038, Sfax – Tunisia

Abstract
The role played by the trunk in gait is relatively unexplored in the literature. Considering that the upper body makes up two third of the body weight and to look at its role as being purely passive is a puzzling hypothesis. This paper investigates the postural strategies to adapt to walking with a backpack load. The aim of the current work is to explore the possibility of using simple models to study the role played by the upper body during specific tasks (in the present reported work walking up an inclined surface with a backpack).

Keywords: Gait analysis, Torso, Optimization

1 INTRODUCTION
The mechanical behavior of the musculoskeletal system and the associated mathematical models have become an important area of biomechanics research [1][2][3][4]. While the existing technologies of motion capture and gait analysis can measure and document major characteristics of human gait and locomotion, they do not have predictive capabilities which are needed to better understand how different anthropometries and states of health or disease affect human locomotion.

Toward this end building a mathematical model that captures the essential features of human locomotion will serve as a platform for understanding how we walk and why our gaits change under different conditions or due to a certain walking impairment.

There seems to be very little work in the literature that investigated the role of the trunk in walking and running. Taking into consideration that the upper part of the body make up nearly two third of the body weight looking at it as being passively carried by the legs seems to be puzzling. The role of the upper limbs in walking may indeed be secondary to that of the lower limbs, but not that much work in the literature have been done to reach the conclusion that their movements are purely passive. We reported in [5,6], that the control of the trunk is important to achieve smooth locomotion and maintain body equilibrium during gait.

The aim of the current work is to explore the possibility of using reasonable models (like the ones used in [7], and [8,9]) to study the role played by the upper body while walking on level ground as well as walking up an inclined surface with a backpack.

The paper using a simple model will seek to show that such models are able to predict postural adaptation of gait to walking with a backpack as reported in the literatures[10-15].

2 MATHEMATICAL MODEL AND PROBLEM FORMULATION
We assume the body to be a hip mass $m_H$ at a position $(x_H, y_H)$ at time $t$, and a trunk mass $m_T$ at a position $(x_T, y_T)$. The trunk or torso is controlled via a torque $r$ between the stance leg and the torso. The legs are massless and symmetric such that only one leg can be considered to study the motion. The fluctuations of the leg length $q(t)$ due to flexion of the hip, knee and ankle are incorporated in a single telescopic axial actuator that carries a compressive time varying force.
The leg has a maximum allowable leg extension, such that \( \sqrt{x^2_H + y^2_H} = R + q(t) \), where \( R \) is the nominal length of the leg. It is assume that during the stance phase, the foot in contact with the ground does not slip, and at most one foot can be in contact with the ground at a given time and that there is no flight phase. The left and right legs have identical force and length profiles.

A gait is characterized by the position and velocity of the hip mass and torso mass, by the step period and by \( F(t) \) and the torque \( \tau(t) \) and the maximum allowed leg extension. We consider \( L \) to be the length of the torso and \( g \) the gravitational acceleration. Using Lagrange formulation the equations of motion are:

\[
(R + q)^2 (m_H + m_T) \ddot{\theta}_1 + (R + q) L m_T \cos(\theta_1 - \theta_2) \ddot{\theta}_2 + 2 (m_H + m_T) (R + q) \dot{q} \dot{\theta}_1 \\
+ (R + q) L m_T \sin(\theta_1 - \theta_2) \ddot{\theta}_2 - (R + q) g (m_H + m_T) \sin(\theta_1) = -\tau \tag{1.1}
\]

\[
(R + q) L m_T \cos(\theta_1 - \theta_2) \ddot{\theta}_1 + L^2 m_T \ddot{\theta}_2 + L m_T \sin(\theta_1 - \theta_2) \dot{q} - L m_T \sin(\theta_1 - \theta_2) (R + q) \dot{\theta}_1^2 + 2 L m_T \cos(\theta_1 - \theta_2) \dot{q} \dot{\theta}_1 - L m_T g \sin(\theta_2) = \tau \tag{1.2}
\]

\[
L m_T \sin(\theta_1 - \theta_2) \ddot{\theta}_2 + (m_H + m_T) \ddot{q} - (m_H + m_T) (R + q) \dot{\theta}_1^2 - L m_T \cos(\theta_1 - \theta_2) \ddot{\theta}_2^2 \\
+ (m_H + m_T) g \cos(\theta_1) = F \tag{1.3}
\]

After scaling the equation of the model using \( M = m_T + m_H, R, \) and \( t_c = \frac{R}{\sqrt{g}} \). The only two free parameters remaining are \( \frac{m_T}{M} \) and \( \frac{L}{R} \). The optimizer will seek solutions as three are varied \( V,D \) and slope. The two remaining parameters namely \( \frac{m_T}{M} \) and \( \frac{L}{R} \) will be fixed.

The governing equations of the motion after non dimensiolizing are

\[
[M] \ddot{Z} + [C(Z,Z)] \dot{Z} + [G(Z)] = [F] \tag{2}
\]

Where,

\[
Z^T = [\theta_1 \quad \theta_2 \quad Q]^T
\]

\( \theta_1, \theta_2, Q \) are the non-dimensional variables.

\[
[M] = \begin{bmatrix}
Q^2 & \frac{L}{R} Q m_T \cos(\theta_1 - \theta_2) & 0 \\
\frac{L}{R} Q m_T \cos(\theta_1 - \theta_2) & L^2 m_T & \frac{L m_T}{R} \sin(\theta_1 - \theta_2) \\
0 & \frac{L m_T}{R} \sin(\theta_1 - \theta_2) & 1
\end{bmatrix}
\]
The humans have an ideal energetic walk. This idealized posture is the result of a natural optimization of the motion of the different part of the body. This shows a close relation between optimization and walking. To highlight this close relation, numerical optimization has been applied to the mathematical model of the human. The optimal solution have cost arbitrary close to zero unless the optimization is constrained. We set the maximum length to $L_{max} = R + q = 1.1R$, representing a maximum increase of 10% of the nominal leg length. The optimization is done for a given step length $d$, using the SQP algorithm of Snopt, a MATLAB toolbox. Further there is no cost associated with standing so we optimize for a given average speed $v$.

Given a particular $d$, assuming that a given step starts with the nominal leg length $(R+q(0) = R)$, we seek the control strategy $\theta$ that minimizes the cost of transport

$$J = \int_{t_0}^{t_{step}} \left| F(t) \dot{q} + |\tau(t)(\dot{\theta}_1 - \dot{\theta}_2)| \right| dt$$

Subject to Eq. (2) and which satisfy the constrains of periodicity: same position and velocity of the torso before and after the stride $\theta_{2f} = \theta_{20}$, $\dot{\theta}_{2f} = \dot{\theta}_{20}$. Same velocity of the hip in both X and Y direction

$$RQ_0 \cos(\theta_{10}) \dot{\theta}_{10} + \dot{Q}_0 \sin(\theta_{10}) = RQ_f \cos(\theta_{1f}) \dot{\theta}_{1f} + \dot{Q}_f \sin(\theta_{1f})$$

$$-RQ_0 \sin(\theta_{10}) \dot{\theta}_{10} + \dot{Q}_0 \cos(\theta_{10}) = -RQ_f \sin(\theta_{1f}) \dot{\theta}_{1f} + \dot{Q}_f \cos(\theta_{1f})$$

3 SIMULATION RESULTS

We run the optimizer for $RM = \frac{m_T}{M}$ representing the ratio between the torso mass (including backpack) and the total mass. Changing this ratio has an effect on the behavior of the different part of the body.

Indeed, the simulation of the model on different mass ratio gives the following results at $V=0.5$ and $D=0.5$ and $L/R=0.46$. 

1020
On level walking changes in trunk posture for different mass ratio are demonstrated in Figure 2. The positive degree means forward inclination. The results showed that increasing $\frac{m_x}{M}$ the backload induced an increase in forward lean of trunk. This is in agreement with the results reported in [10-12]. One possible role for forward lean (as explained in [10] is to maintain the Center of mass of the entire biped ($\text{COM}_{\text{TOT_SYS}}$) in a position such “the vector from the $\text{COM}_{\text{TOT_SYS}}$ to the ankle in the sagittal plane remains constant across different loads”. The initial backward lean did not show a consistent trend with the increased load, however the rate of leaning forward (\(\dot{\theta}_2\)) showed a consistent increase as the load is increased. We did not in the current reported results looked at how the load placement (a way or close to the hip) will affect the results.

Figure 2. Torso angle for different mass ratio

Figure 3. Normalized leg extension for different mass ratio
Figure 3 shows that the fluctuation of the leg length is decreasing at the beginning of the step then increases at the end. This can be explained by the contraction of the muscles and the flexion of the knee and the ankle before push off and at heel strike [16]. Further, the stance leg behaves like an inverted pendulum swinging about the stance foot for most of the step period. Figure 3 shows as well that for high values of RM the leg extension increases. To carry the additional load coming from the mass torso the leg bends more.

![Graph](image.png)

**Figure 4.** Evolution of the cost of transport with the mass ratio

High value of RM could be considered as an additional load that the person carries on his back. To carry this load the person need more energy and this is highlighted by figure 4 that shows that for high value of RM the value of the COT (Cost of transport) increases. Although there appears to be a linear relationship between the metabolic cost and the amount of load carried, it is unknown if load carrying decreases the stability of the gait pattern.

4 CONCLUSIONS

Torso role in walking is yet to attract attention; this due to the fact that this motion is relatively small and is generally thought to be passive. Recent studies however, highlights the importance of torso motion in walking. Its kinematics has been associated with age-related changes and with maintenance of dynamic stability in elderly individuals. Trunk motion is of large clinical interest, in patients with spinal pathologies, Parkinson or cerebral palsy diseases. The central aim of this on-going work is to use optimization to predict and explore the significance role that the trunk has in walking especially when walking uphill or downhill or when speeding up or slowing down. These are preliminary results with a very simple model. The long term objectives are:

(a). To explore the role played by the trunk in locomotion.
(b). To use trunk motion to detect gait parameters.
(c). To develop simple models that have predictive capabilities to use to analyze pathological gaits and judge certain treatment

Acknowledgement(s)

This work was a result of a research project supported by Sultan Qaboos University under code number IG/ENG/MIED/12/01.
REFERENCES


[12]. Cottalorda, Jérôme, MD; Bourelle, Sophie, MD; Gautheron, Vincent, MD, “Effects of Backpack Carrying in Children”, Orthopedics, Vol. 27, No. 11(2004), pp. 1172-1175


Influence of muscle recruitment criteria on joint reaction forces during human gait

Florian Michaud*, Urbano Lugris*, Ye Ou#, Javier Cuadrado*, Andres Kecskemethy#

* Lab. of Mechanical Engineering
University of La Coruña
Escuela Politecnica Superior
Mendizabal s/n, 15403 Ferrol, Spain
florian.michaud@udc.es

# Institute of Mechanics and Robotics
University of Duisburg-Essen
Lotharstr. 1, 47057 Duisburg,
Germany
andres.kecskemethy@uni-due.de

ABSTRACT
This paper addresses the comparison of several muscle recruitment criteria and their effect on the calculated joint reaction forces at hip, knee and ankle level during gait. Both the kinematics and the ground reactions are experimentally obtained from a female subject and applied to a three-dimensional skeletal human model featuring 43 muscles in the right leg. An inverse dynamic analysis provides the histories of the joint drive torques. Then, static –four criteria–and physiological approaches are used to estimate the muscle forces from which the joint reaction forces are calculated and can then be compared. It is concluded that significant differences are observed in the results. Moreover, some issues which came across in the course of the investigation are pointed out and discussed.

Keywords: Biomechanics, Gait, Muscle forces, Joint reactions, Muscle recruitment criteria.

1 INTRODUCTION
Determination of muscle forces during gait (or any other exercise) is of great interest to extract the principles of the central nervous system (CNS) control [1] (assessment of pathological gait from muscular activation abnormalities, diagnosis of neuromuscular disorders), or to estimate the loads on bones and joints [2] (prevention of injuries in sports, surgical planning to reconstruct diseased joints). The invasive character of in vivo experimental measurements, and the uncertain relation between muscle force and EMG, makes computer modeling and simulation a useful substitutive approach [3].

The fundamental problem is that there are more muscles serving each degree of freedom of the system than those strictly necessary from the mechanical point of view, which implies that, in principle, an infinite number of recruitment patterns are acceptable. This problem is often referred to as the redundancy problem of the muscle recruitment [4] or the force-sharing problem [5]. Experimental studies [6] and EMG collections [7] suggest that a specific strategy of muscle coordination is chosen by the CNS to perform a given motor task.

A popular mathematical approach for solving the muscle recruitment problem is the optimization method, which can be associated to inverse or forward dynamics [8]. These methods minimize or maximize some criterion (objective function or cost function) which reflects the mechanism used by the CNS to recruit muscles for the movement considered. The proper cost function is not known a priori, so the adequacy of the chosen function must be validated according to the obtained results [9]. Many criteria have been proposed in the literature to predict muscle forces. However, according to Daniel [10], the choice of the optimization criterion does not influence the hip reaction force in the inverse dynamic analysis.

In this work, the gait of a female subject has been analyzed and the muscle forces of the right leg have been obtained by applying both static (four different criteria) and physiological approaches. The objective is to compare the joint reaction forces at hip, knee and ankle level and to observe whether significant discrepancies appear depending on the selected method.
The remaining of the paper is organized as follows. Section 2 presents the experiment carried out and the human model used. Section 3 addresses the application of the static optimization approach to get the muscle forces, describes the four muscle recruitment criteria that have been compared in the work, and shows the obtained joint reaction forces. Section 4 explains the physiological optimization approach and provides the corresponding joint reaction forces. Finally, the conclusions of the paper are drawn in Section 5.

2 EXPERIMENT AND MODEL

The subject selected to perform the experiment is a healthy adult female, 28 years old, mass 50 kg and height 1.67 m. She walks on a walkway featuring two embedded force plates (AMTI, AccuGait sampling at 100 Hz). The motion is captured by 12 optical infrared cameras (Natural Point, OptiTrack FLEX:V100 also sampling at 100 Hz) that compute the position of 37 optical markers.

![Figure 1. 3D human model and detail of muscles on the right leg.](image)

The human body is modeled as a 3D multibody system formed by rigid bodies, as shown in Figure 1 (left and center). It consists of 18 anatomical segments: two hindfeet, two forefeet, two shanks, two thighs, pelvis, torso, neck, head, two arms, two forearms and two hands. The segments are linked by ideal spherical joints, thus defining a model with 57 degrees of freedom. The global axes are defined as follows: \(x\)-axis in the antero–posterior direction, \(y\)-axis in the medio–lateral direction, and \(z\)-axis in the vertical direction. The computational model is defined with 228 mixed (natural + angular) coordinates. The subset of natural coordinates comprises the three Cartesian coordinates of 22 points, and the three Cartesian components of 36 unit vectors, thus making a total of 174 variables. The points correspond to the positions of all the spherical joints (white dots in Figure 1, left and center), along with points of the five distal segments - head, hands and forefeet- (black dots in Figure 1, left and center). Each one of the 18 bodies is defined by its proximal and distal points, plus two orthogonal unit vectors aligned at the antero–posterior and medio–lateral directions, respectively, when the model is in a standing posture. The remaining 54 variables are the 18 sets of 3 angles that define the orientation of each body with respect to the inertial frame.

The geometric and inertial parameters of the model are obtained, for the lower limbs, by applying correlation equations from a reduced set of measurements taken on the subject, following the procedures described in [11]. For the upper part of the body, data from standard tables [8] is scaled according to the mass and height of the subject. In order to adjust the total mass of the subject, a second scaling is applied to the inertial parameters of the upper part of the body.
The kinematic information of the motion is obtained from the trajectories of the 37 markers attached to the subject’s body (red dots in Figure 1, left), which are captured at 100 Hz frequency by means of the 12 infrared cameras. Position data are filtered using an algorithm based on Singular Spectrum Analysis (SSA) and the natural coordinates of the model are calculated using algebraic relations. Afterwards, a minimization procedure ensures the kinematic consistency of the natural coordinates. From that information, the histories of a set of 57 independent coordinates -as many as the system degrees of freedom- formed by the Cartesian coordinates of the position vector of the lumbar joint and the 18 x 3 angles that define the absolute orientation of each body, are kinematically obtained and approximated by using B-spline curves. Analytical differentiation yields the corresponding velocity and acceleration histories. More detail about the treatment of the captured data can be found in [12].

The matrix-R formulation [13] is applied to obtain the ground reactions and joint drive torques along the motion. Measurements from the force plates are just used to overcome the indeterminacy in the distribution of ground reactions during the double support phase. Therefore, after the analysis, a set of joint drive torques and external reactions is available which is consistent with the corresponding motion.

In the right leg of the model, 43 muscles have been considered (Figure 1, right). The properties of the muscles have been taken from OpenSim [14], which are defined for the OpenSim reference model. A scale factor is derived for each segment by comparing its dimensions with those of the reference model. This factor is applied to obtain the corrected location of insertion points in the segment. Then, lengths of muscles are calculated and compared with their counterparts in the reference model, thus yielding a scale factor for each muscle. This scale factor is applied to muscle parameters as the tendon slack length and the optimal muscle fiber length. However, no recommendation has been found in the literature on how to scale the muscle maximum isometric force, which could be expected to significantly vary among different subjects.

3 STATIC OPTIMIZATION

The first approach considered is static optimization. Since only muscles in the right leg have been modeled, joint drive torques at the right hip, knee and ankle should be reproduced by the muscle forces. The following optimization problem is stated,

$$\min C$$

subject to

$$J^\top F = Q$$

$$0 \leq F_i \leq F_{i,0} \quad i = 1,2,\ldots,m$$

where $C$ is the cost function, $Q$ is the vector of joint drive torques at the right leg (where the force-sharing problem is addressed), $F$ is the vector of muscle forces, $J$ is the Jacobian whose transpose projects the muscle forces into the joint drive torques space, and $F_{i,0}$ is the maximum isometric force of muscle $i$, with $m$ the number of muscles (in this case, 43).

Regarding the cost function $C$, four cases have been considered and compared, whose mathematical formulations are shown in Table 1:

I) Sum of the squares of muscle forces.

II) Sum of the squares of proportional muscle forces.

III) Sum of muscle stresses, with $A_i$ the cross sectional area of muscle $i$.

IV) Largest relative muscle force.
Before showing the results, there is an issue which deserves to be mentioned. Vector $Q$ in (1) gathers the joint drive torques at the right hip, knee and ankle. Since there are bi-articular muscles, i.e. muscles spanning more than one joint, the optimization problem cannot be carried out on a joint-by-joint basis, but instead all the joints should be taken into account at the same time. On the other hand, it has been said when describing the human model that spherical kinematic pairs have been considered for all the joints. This means that three joint drive torques are obtained at each joint from the inverse dynamic analysis. However, not all of them are due to the actuation of muscles. For example, it is clear that the abduction/adduction torque at the knee is not provided by muscles, but rather by other joint structures as condyles and ligaments, being more a reaction moment than a drive torque. Therefore, the following joint drive torques have been selected in this work to form vector $Q$: the three torque components at the hip, the flexion/extension torque at the knee, and the plantarflexion/dorsiflexion and abduction/adduction torques at the ankle. A discussion on how the modeling of the joints and the torques considered in the optimization affect to the results can be found in [15].

Based on these assumptions, the joint reaction forces were computed at the right hip, knee and ankle for the four mentioned criteria, which are plotted in Figure 2.

Conversely to what was suggested in [10], results show that different criteria lead to notably different values of the joint reaction forces. More specifically, variations of the maximum force of about 35% at the hip, 70% at the knee and 20% at the ankle can be observed in Figure 2.

### 4 PHYSIOLOGICAL OPTIMIZATION

In constrast to the static optimization, the so-called physiological static optimization takes muscle dynamics into account by introducing dynamic muscle force constraints [16]. This method applies static optimization techniques at each time-point but prescribes minimal and maximal constraints for the muscle forces by extrapolating the force values from the previous time-point using feasible muscle activation values. In this way, the optimization process remains efficient, but muscle dynamics are considered.

In this approach, the following optimization problem is stated,
\[
\min \sum_{i=1}^{m} \left( \frac{F_i}{F_{i,\text{max}}} \right)^2 \\
\text{subject to } J^T F = Q \\
F_{i,\text{min}} \leq F_i \leq F_{i,\text{max}} \quad i = 1, 2, \ldots, m
\]

where all the symbols have the same meaning as in (1), and \( F_{i,\text{min}}, F_{i,\text{max}} \) are, respectively, the minimum and maximum admissible forces for muscle \( i \) at the corresponding time-point. In what follows, the way to determine such force limits is explained.

If the Hill's muscle model is used [17], the states of muscle \( i \) are denoted by the vector (index \( i \) is dropped for simplicity),

\[
x = \begin{bmatrix} a \\ F \end{bmatrix}
\]

where \( a \) is the muscle activation and \( F \) is the musculotendon force. The Hill's muscle first-order differential equations are,

\[
\dot{x} = \begin{bmatrix} \dot{a} \\ \dot{F} \end{bmatrix} = \begin{bmatrix} f_1(a, u) \\ f_2(a, F, l, v) \end{bmatrix} = f(x, u, l, v)
\]

being \( u \) the muscle excitation, \( l \) the musculotendon length and \( v \) the musculotendon velocity.

If the states are given at a certain time \( t \), the minimum and maximum state variables at time \( t + \Delta t \) can be computed by setting the neural input \( u \) to its minimum (\( u = 0 \)) and maximum (\( u = 1 \)) possible values during the time interval \( \Delta t \).

\[
x_{\text{min}}(t + \Delta t) = x(t) + \int_t^{t+\Delta t} f(x, u = 0, l, v) dt \\
x_{\text{max}}(t + \Delta t) = x(t) + \int_t^{t+\Delta t} f(x, u = 1, l, v) dt
\]

The two integrations in (5) have been performed by using numerical integrator \( \text{ode23} \) from Matlab. Values of \( l \) and \( v \) inside the time interval \( \Delta t \) are obtained by linear interpolation of their corresponding values at times \( t \) and \( t + \Delta t \). The solution of (5) provides the limits \( F_{i,\text{min}}, F_{i,\text{max}} \) for muscle \( i \). This process must be repeated for all the muscles.

It must be noted that the lowest activation at \( t + \Delta t \) is not always obtained for \( u = 0 \). In the long term, the activation converges to the excitation value if the latter remains constant. However, for small \( \Delta t \) values, an excitation higher than 0 can lead to a lower activation at \( t + \Delta t \). Therefore, the \( F_{i,\text{min}} \) used for the optimization is not always guaranteed to be the smallest possible, although the error remains under 2.5% of the maximum activation.

Once the force limits for all the muscles have been determined, the optimization problem (2) can be solved, thus yielding the muscle forces \( F_i, i = 1, 2, \ldots, m \) for time \( t + \Delta t \). At this point, an iteration process for each muscle must be run in order to find out the (assumed constant) excitation value \( u \) during the time interval \( \Delta t \) that leads to the obtained muscle force \( F_i \) at time \( t + \Delta t \). To that end, different values of \( u \) (index \( i \) is dropped again) are tried until the bottom part (that affecting the force; see (3)) of the following equation is satisfied,
\[
\mathbf{x}(t + \Delta t) - \mathbf{x}(t) - \int_t^{t+\Delta t} \mathbf{f}(\mathbf{x}, u, l, \nu) dt = 0
\] (6)

Function \textit{fsolve} from Matlab has been used for the iteration process, starting with initial guess \( u = 1 \). The bottom part of (6) is integrated for each value of \( u \) provided by \textit{fsolve}, until the resulting muscle force falls within a certain tolerance of the force obtained in the optimization (2). The companion muscle activation can be then obtained from the upper part of (6), being the activation at time \( t + \Delta t \).

So far, it has been assumed that the muscle states are known at time \( t \) in order to move to time \( t + \Delta t \). Therefore, a particular procedure must be followed for the initial conditions, i.e. at time \( t = 0 \). For that time, it is supposed that muscle velocity is zero, \( v_m = 0 \), for all the muscles, which implies that the force-velocity relationship of the Hill's muscle model is equal to one, \( f_c(v_m = 0) = 1 \). To determine the initial muscle forces, the optimization problem (2) must be solved, being the force limits \( F_{i,\text{min}} \) and \( F_{i,\text{max}} \) the ones obtained by considering the minimum and maximum muscle activations, respectively, \( a = 0 \), \( a = 1 \). According to Figure 3, the force equilibrium demands (index \( i \) is dropped again),

\[
F = F_T = (F_{PE} + F_{CE}) \cos \alpha
\] (7)

being \( F \) the musculotendon force, \( F_T \) the tendon force, \( F_{PE} \) the force of the parallel element, \( F_{CE} \) the force of the contractile element, and \( \alpha \) the pennation angle.

![Figure 3. Hill's muscle model.](image)

Dependencies of the previous force magnitudes are as follows: \( F_T = F_0 f_f(l - l_m \cos \alpha) \), \( F_{PE} = F_0 f_f(l_m) \), \( F_{CE} = a_0 f_f(l_m) f_c(v_m = 0) \). Therefore, in the second equality (7), only the muscle length \( l_m \) is unknown, which means that it can be worked out, although in an iterative way, since the actual expressions are rather involved. Function \textit{fsolve} from Matlab is used again for this purpose, taking the optimal muscle fiber length as initial guess.

The described problem must be solved twice for each muscle, for \( a = 0 \) and \( a = 1 \), respectively, thus providing, via the first equation in (7), the limits \( F_{i,\text{min}} \) and \( F_{i,\text{max}} \) required for the optimization problem (2) at the initial time, \( t = 0 \). Once the optimization problem has been solved, the initial force of each muscle \( F_f \) is obtained, and the corresponding initial activation \( a_i \) can be derived by equaling the leftmost and rightmost equality side in (7).

A more detailed explanation of this method can be found in [18].

The joint reaction forces obtained at the right hip, knee and ankle by application of the described physiological method are plotted in Figure 4, along with the results obtained in the previous Section for the static methods, so that comparison is easily established.
Figure 4. Joint reaction forces at hip, knee and ankle for static and physiological methods.

As it can be seen in Figure 4, the physiological method leads to even greater reaction forces that its static counterparts, thus confirming the influence of the calculation approach in the results.

5 CONCLUSIONS

The acquired gait of a female subject has been analyzed through a three-dimensional human model featuring 43 muscles in the right leg. Static optimization with four difference muscle force-sharing criteria and static-physiological optimization have been applied to estimate the histories of muscle forces, based on which, hip, knee and ankle reaction forces have been derived and compared for the five different calculation methods. Results show that the maximum joint reaction forces can significantly vary at hip, knee and ankle depending on the method selected to estimate the muscle force distribution.

It should be remarked that there are some relevant issues in the muscle force estimation process that may require further attention, like the influence of the human model degrees of freedom and the joint drive torques considered in the optimization, or the scaling of maximum isometric force of muscles.

ACKNOWLEDGMENT

The support of this work by the Spanish Ministry of Economy and Competitiveness (MINECO) under project DPI2012-38331-C03-01, conﬁnanced by the European Union through EFRD funds, is greatly acknowledged.

REFERENCES


Inverse Dynamics Analysis of the Driver-Seat-Steering System using a Multibody Human Model

Masatoshi Hada, Ei-ichi Yasuda, Takao Kobayashi, Takahiro Yamaguchi, Yoshikazu Hattori

TOYOTA Central Research & Development Labs., In.
Nagakute, Aichi 480-1192, Japan
hada@vdlab.tytlabs.co.jp

ABSTRACT
This work presents a method for analyzing the interaction between driver and vehicle by combining multibody inverse dynamics and experimentally obtained data from different sources. The driver is modeled as 23 rigid bodies and joints with 64 degrees of freedom. The movements of the driver are collected by motion capture to measure the three dimensional positions of reflective markers. Steering forces are measured by a steering wheel unit with six axis transducers on the spokes. To extract the net force exerted by the driver’s hands, inertia forces due to gravity, rotation of the steering wheel, and vehicle dynamics are compensated. Pedaling forces are measured in the same way as the steering forces. Pressure distributions between the driver and the seat are measured by capacitive sensor mats. Equivalent force and torque are calculated based on measured pressure and the shape of the seat surface. All the measured forces and movements of the vehicle are considered as external forces and additional gravitational effect in the equation of driver motion. To investigate the effects of vehicle motion, seat side support, and seat shoulder support on the lumbosacral joint torque, a skillful driver’s slalom and lane-change maneuvers were measured on a six axis motion generator. The inverse dynamics were calculated by scaling the measured vehicle motion and seat contact forces. The results suggest that joint torque in the yaw direction is used dominantly for turning the steering wheel, and that shoulder support force has more effect on the steering operation than side support force.

Keywords: Inverse dynamics, Biomechanics, Driver vehicle interaction.

1 INTRODUCTION
Seeing a frontal view through the eyes in her/his head, a driver controls a vehicle by skillfully contracting her/his muscles around skeletal bones. Muscle forces acting in cooperation generate joint torques and these are transmitted to the steering wheel and the pedals of the vehicle. At the same time, seat reaction forces and inertial forces due to vehicle dynamics are exerted upon the driver’s body. In that context, as shown in Figure 1, the driver in a vehicle can be regarded as a multibody system subjected to various external forces.
A large number of studies have focused on driver modeling to investigate the various aspects of a driver-vehicle closed loop. Cole [1] and Müller [2] take a control theoretic modeling approach considering neuromuscular dynamics. Unlike conventional driver models in which the steering angle is the input to the vehicle dynamics, their models use a path-following feedback loop and a comprehensive neuromuscular minor loop to describe a driver’s response to steering torque.

Mehrabi and McPhee [3], on the other hand, take a more biomechanical approach, in which musculoskeletal and neuromuscular models are combined with a general purpose multibody dynamics code. Explicitly described muscle activation and contraction dynamics take energy consumption and muscle fatigue into consideration. Recently, Masarati [4] developed a multibody biomechanical model of a pilot, including the musculoskeletal system, to simulate involuntary inputs by pilots.

In this work, we take a practical, experiment-based and inverse dynamics approach to investigate the interaction between driver and vehicle. Both of the above mentioned models need to obtain mechanical and descriptive characteristics in terms of driver motion. Although all the external forces, and the motion of the driver and the vehicle must be measured, our proposed method provides significant insights for developing seat characteristics, and a more realistic driver model.

2 MEASUREMENT OF MOTION AND FORCE

2.1 Six axis motion generator

A six axis motion generator, see Figure 2, was used to expose a driver to simulated vehicle motion. Three pairs of pantograph-like links are actuated by six motors mounted on the lower arm of the links. The cockpit can move six dimensionally following the prescribed trajectory given on a reference frame of the generator.

The motors are controlled to follow the desired lower arm angular velocity \( \dot{\alpha} \in \mathbb{R}^6 \), which is obtained from the desired reference frame velocity \( \dot{x}_r \in \mathbb{R}^3 \) and time derivative of the desired roll, pitch, yaw angle \( \dot{\Theta}_r = [\dot{\phi}, \dot{\theta}, \dot{\psi}]^T \in \mathbb{R}^3 \). Both velocity and time derivative of the configuration defined in the fixed frame \( \Sigma_f \) in Figure 2. Using Jacobian matrix \( J_\alpha \in \mathbb{R}^{6 \times 6} \), the relation between \( \dot{\alpha} \) and \( [\dot{x}_r, \dot{\Theta}_r]^T \) can be described as

\[
\dot{\alpha} = J_\alpha \dot{x}_r ,
\]

\[
\dot{x}_r = [A_r^T, G_{wv}] \begin{bmatrix} \dot{x}_r \\ \dot{\Theta}_r \end{bmatrix},
\]

where \( \dot{x}_r \in \mathbb{R}^6 \) is the velocity and angular velocity of \( \Sigma_r \) moving with the floor shown in Figure 3.
\( A_r \) is the rotation matrix between \( \Sigma_f \) and \( \Sigma_r \). \( G_{wvu} \) is the tangent operator for the Cardan angle to describe rotation. They are respectively described as follows:

\[
A_r = \begin{bmatrix}
\cos \theta \cos \psi & \sin \phi \sin \theta \cos \psi - \cos \phi \sin \psi & \cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi \\
\cos \theta \sin \psi & \sin \phi \sin \theta \sin \psi + \cos \phi \cos \psi & \cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi \\
-\sin \theta & \sin \phi \cos \theta & \cos \phi \cos \theta
\end{bmatrix}
\]  

\( G_{wvu} = \begin{bmatrix}
0 & -\sin \psi & \cos \theta \cos \psi \\
0 & \cos \psi & \cos \theta \sin \psi \\
1 & 0 & -\sin \theta
\end{bmatrix}
\]  

In a preliminary experiment, we obtained a flat frequency response up to 13 Hz. This seemed to be sufficient to investigate the motion of a driver under the first eigenfrequency of the whole body, between 4 and 5 Hz [5]. Also, the cockpit has a motor to apply steering reaction torque according to operation of the driver.

### 2.2 Motion capture

The movements of the driver’s segments are collected by standard infrared cameras (VICON T160, Vicon Motion Systems Ltd.) to measure the three dimensional positions of reflective markers on the driver. A total of 60 reflective markers on the driver and 5 markers on the frame of the motion generator are measured by 14 cameras in this study.

### 2.3 Force/torque transducers

Figure 4 shows the force/torque transducers (Mini-45, Delta, ATI) on the spokes of the steering wheel and behind the pedals. To measure the steering force applied by the right and left hands separately, the grip part of the steering wheel unit forms a single supported beam. Inertial forces due to gravity, rotation of the steering wheel, and vehicle dynamics must be eliminated to extract the net force exerted by the driver. Figure 5 shows the schematic process of inertia compensation in this study. The measured steering force and torque, foot force is denoted as \( F_{\text{steer}} \in \mathbb{R}^{12} \) and \( F_{\text{pedals}} \in \mathbb{R}^{12} \).
2.4 Seat pressure distribution

Pressure distributions between the driver and seat are measured by capacitive sensor array mats (Xsensor, LX 48 × 48, LX 40 × 40). The mats are placed on the surface of the seat back and the seat cushion. As shown in Figure 6a, the three dimensional positions of every fourth cell are measured by the motion capture system. The positions of the neighboring cells are then estimated by using bilinear interpolation (see Figure 6b). Next, the cells are divided into 8 areas to define the contact between the driver and the seat. Assuming that the output of each cell is the force along the normal vector of the cell surface, equivalent force $F_{\alpha_n}$ and torque $\tau_{\alpha_n}$ applied at the center of pressure in the area $\alpha_n, n = 1, 2, ..., 8$ can be calculated as follows:

\[
F_{\alpha_n} = \sum_{i,j \in \alpha_n} f'_{i,j} \tag{5}
\]

\[
\tau_{\alpha_n} = \sum_{i,j \in \alpha_n} (x_{i,j} - x_{\alpha_n}) \times f'_{i,j} \tag{6}
\]

where $x_{i,j}$ describes the position of the $(i, j)$ th cell of the mat, $x_{\alpha_n}$ is the center of pressure in the area $\alpha_n$, and $f'_{i,j}$ is the measured force applied at the $(i, j)$ th cell. The measured seat force and torque $F_{\text{seat}} \in \mathbb{R}^{48}$ can be denoted as

\[
F_{\text{seat}} = [F_{\alpha_1}^T, \tau_{\alpha_1}^T, F_{\alpha_2}^T, \tau_{\alpha_2}^T, ..., F_{\alpha_8}^T, \tau_{\alpha_8}^T]^T \tag{7}
\]

In this study, a non-deformable seat was used so that the positions of the cells are fixed in the measurements.

3 ANALYSIS OF DRIVER-VEHICLE INTERACTION

3.1 Multibody human model

Figure 7a shows the multibody human model used in this study. The human body is modeled as 23 rigid bodies. Each rigid body has a local coordinate frame called a marker to define the position of the joint, the constraint, and the corresponding reflective markers on the driver for the motion capture system. All the joints have only rotational degrees of freedom, and a free joint is defined between the pelvis and the ground, having a total of 64 degrees of freedom in the human model. The length and mass of each segment is scaled based on the measured marker position of the participant. The center of mass and inertia of each rigid body are defined with reference to the biomechanics literature, such as [7].

The shoulder complex is a closed chain mechanism in which the humeral head is positioned by a closed chain formed by the thorax, scapula, and clavicle [8]. The scapulothoracic joint is modeled as a constraint between the thorax and scapula. This constraint means that an axis of the markers defined in the scapula is always orthogonal to the surface of the virtual ellipsoid that describes the surface of the thorax, as shown in Figure 7b.
3.2 Inverse kinematics

Let \( q \in \mathbb{R}^{64} \) denote the posture of the human model at time \( t \); then the positions of the reflective markers on the model \( X \in \mathbb{R}^{3m} \), where \( m \) is the number of the markers, can be described as the following nonlinear function:

\[
X = f(q)
\]  

(8)

Differentiating both sides of Equation (8) with respect to time gives

\[
\dot{X} = J(q) \dot{q}
\]  

(9)

where \( J(q) \in \mathbb{R}^{3m \times 64} \) is the Jacobian matrix of \( X \) with respect to \( q \).

Let us consider reducing the error \( E = X_m - X \in \mathbb{R}^{3m} \) between \( X \) and \( X_m \), the position of the reflective markers on the driver. Optimum value \( q^* \) can be obtained by repeating the following updating rule until convergence:

\[
q_{k+1} = q_k + \Delta q
\]  

(10)

\[
\Delta q = \varepsilon J^+(q_k) E_k
\]  

(11)

where \( k \) is the number of iterations, \( \varepsilon \) is the positive updating factor, and \( J^+ \in \mathbb{R}^{64 \times 3m} \) is the pseudo inverse matrix of \( J \). Applying Equations (10) and (11) to the measured motion capture data, a time series of the driver’s joint angle and pelvis posture \( q(t) \) \( (t = 0, 1, ..., t_f) \) can be obtained as shown in Figure 8a. Furthermore, appropriate differentiation gives the velocity \( \dot{q}(t) \) and acceleration \( \ddot{q}(t) \).

Once \( q \) and \( \dot{q} \) are obtained, the velocity and angular velocity of any point of the human model \( x \) can be calculated by

\[
\begin{bmatrix}
\dot{v} \\
\omega
\end{bmatrix} = J_x(q) \dot{q}
\]  

(12)

where \( v \in \mathbb{R}^3 \) and \( \omega \in \mathbb{R}^3 \) are the velocity and angular velocity of \( x \), respectively, and \( J_x(q) \in \mathbb{R}^{6 \times 64} \) is the Jacobian matrix of \( x \) with respect to \( q \). Figure 8b shows an example of the floor, head, thorax, and pelvis velocities and angular velocities.

3.3 Instantaneous screw axis

According to the Euler’s theorem of rotation, the instantaneous screw axis can be obtained as follows [9]:

\[
x_M = \frac{\omega}{|\omega|^2} v + \left( I - \frac{\omega \omega^T}{|\omega|^2} \right) x
\]  

(13)

\[
u = \frac{1}{|\omega|} \omega^T v
\]  

(14)
3.4 Inverse dynamics

The equation of motion for the human model is formulated as the following differential algebraic equations (DAEs):

\[
\begin{bmatrix}
M & G^T
\end{bmatrix}
\begin{bmatrix}
\ddot{q} \\
\lambda
\end{bmatrix}
= 
\begin{bmatrix}
Q - g - h - J_e^T F_e
\end{bmatrix}
\gamma
\]  

(15)

where \( I \in \mathbb{R}^{3\times3} \) is the unit matrix, \( u \) is the pitch of the screw, and \( x_M \) is the closest point on the axis of rotation of \( x \). Figure 9 shows the instantaneous screw axis of the pelvis center of mass, with a line through \( x_M \), scaled by the magnitude of the angular velocity \( \omega \). It can be seen that the distance between the axis and the pelvis depends on the component of \( v \) which is orthogonal to \( \omega \).

Successively substituting measured motions \( q, \dot{q}, \ddot{q} \) and forces \( F_e \) into the formulated Equation (15), generalized force \( Q \) is calculated by using a projection method [11] as

\[
P \dot{\ddot{q}} = P( Q - g - h - J_e^T F_e )
\]  

(17)

where \( P \) is the matrix that describes the projection of \( G \) onto null space. In this method, it is not necessary to calculate the constraint force \( \lambda \). Therefore, the joint torque of the human body can be obtained with the minimum set of inputs. Figure 8c shows an example of estimated joint torque in the left shoulder and the lumbosacral joint.

Figure 8. Inverse kinematics and inverse dynamics

Figure 9. Instantaneous screw axis of the pelvis
4 APPLICATION

4.1 Experimental protocol

We conducted an experiment to demonstrate the effectiveness of the proposed analysis. After obtaining informed consent, an expert male driver in his 50s joined in the experiment. The height and weight of the participant were Japanese 50 percentile, 1.76 m and 76 kg, respectively.

As shown in Figure 10a, the participant sat in the cockpit of the motion generator with a general passenger car layout consisting of steering wheel, pedals, and seat. The steering wheel inclination angle was 20 degrees and the torso angle of the participant was 23 degrees. The participant was instructed to put his hands and feet on the grip parts of the steering wheel unit and the pedals, respectively.

In order to give highly repeatable vehicle acceleration input to the participant, the trajectory of the motion generator was calculated in advance by the following equation of vehicle motion [12], supposing a 0.4 Hz slalom and a 0.75 Hz lane-change maneuver at 100 km/h:

\[ m_v V \left( \frac{d\beta}{dt} + \gamma \right) = -2K_f \left( \beta + \frac{l_f}{V} \gamma - \delta \right) - 2K_r \left( \beta - \frac{l_r}{V} \gamma \right) \]

\[ l_i \frac{d\gamma}{dt} = -2K_f \left( \beta + \frac{l_f}{V} \gamma - \delta \right) l_f + 2K_r \left( \beta - \frac{l_r}{V} \gamma \right) l_r \]

\[ \frac{d^2\phi}{dt^2} = -2\zeta_\phi \omega_\phi \frac{d\phi}{dt} - \omega_\phi^2 \phi + K_\phi \omega_\phi^2 V \left( \frac{d\beta}{dt} + \gamma \right) \]

where \( \beta \) is slip angle, \( \gamma \) is yaw angular velocity, and \( \phi \) is roll angle. \( V \) is velocity of the vehicle, and \( m_v \) and \( I_i \) are mass and yaw moment of inertia, respectively. \( \delta \) is the steering angle, and \( l_f \) and \( l_r \) are the front and rear yaw moment of inertia, respectively.
and $l_r$ are the distance between the vehicle center of mass and the front/rear wheels, respectively. $K_f$ and $K_r$ are the front and read cornering stiffnesses, $K_\phi$ is roll gain, $\zeta_\phi$ is roll damping ratio, and $\omega_\phi$ is roll natural angular frequency. The participant was exposed to the motion of the vehicle for 30 seconds after 10 seconds of transient input. The input steering wheel angle and response of the vehicle are shown in Figure 11. A positive sign for the steering wheel angle means a right turn, and the resulting lateral acceleration of the vehicle was $1.8 \text{ m/sec}^2$. Let it be noted that the steering reaction torque was applied based on the lateral force of the tires and an assist algorithm. Both the desired and actual steering wheel angles were imposed on a computer-generated front view, which was projected onto a screen 6.5 m in front of the motion generator (see Figure 10b). During the experiment, the participant rotated the steering wheel to match the desired angle shown on the screen. In accordance with the peaks of the desired steering wheel angle, a beep cyclically sounded to make the maneuver easier to achieve. Figure 12 shows a comparison of the desired and measured steering wheel angles between 25 and 30 seconds after starting the experiment.

The movements and applying forces of the participant are all collected at 100 Hz. The seat pressure distributions are collected at 50 Hz and interpolated with respect to time during data processing.

### 4.2 Angular velocity of driver’s body

Figure 13 shows the roll angular velocity of the head, the thorax, and the pelvis of the driver and the floor of the cockpit with respect to the ground. The yaw angular velocities of these parameters are plotted in Figure 14.

Figure 13 shows that the pelvis rolls to the outer side of a turn, whereas the thorax and the head conversely roll to the inner side of a turn. Makita et al. [13] and Yokoyama et al. [14] obtained similar results in their vehicle experiments at a higher lateral acceleration range between 3.0 and $6.0 \text{ m/sec}^2$. The results of this study indicate that the same kind of motion can be observed, even in a lower lateral acceleration range.

Figure 14, on the other hand, shows that the largest yaw angular velocity can be seen in the head, followed by the thorax and the pelvis, in that order. In terms of phase difference, the head leads the pelvis, whereas the thorax follows. The head yaw motion implies that the participant continued to focus on the center of the screen even although the motion generator moved laterally, so as to match the steering wheel angle to the desired angle. Yokoyama et al. also indicated that the upper body of a driver twisted to push against the outer side of a seat back during a turn [14].

**Figure 13.** Roll angular velocity during slalom

**Figure 14.** Yaw angular velocity during slalom
4.3 Instantaneous screw axis and longitudinal forces

To investigate the observed upper body motion in more detail, the instantaneous screw axis of the thorax and the pelvis with respect to the floor, and the relative screw axis between the thorax and the pelvis are calculated by substituting the measured lateral velocity, roll, and yaw angular velocity for Equations 13 and 14.

Figure 15 depicts the above mentioned three axes at the time when the largest magnitude of the relative angular velocity between the thorax and the pelvis is observed during left and right turns. It is found that the screw axis of the thorax passes close to the center of mass of the head, not the thorax, in both cases. Apparently the thorax moves in a way that decreases the displacement of the head. This indicates that the participant unconsciously keeps his head as stable as possible. In addition, the relative screw axis between the thorax and pelvis moves close to the center of mass of the driver’s bodies above the knee. This indicates that the participant makes an effort to maintain his posture despite the fact that lateral inertial forces are applied to his whole body.

In order to account for the above matters from a mechanical point of view, the longitudinal components of the steering forces and the shoulder supporting forces shown in Figure 16 are extracted from the measured data and plotted in Figure 17. From the peak values of the steering forces, it can be seen that the participant pushes the steering wheel forward using the hand on the outer side of a turn. The longitudinal steering force of the inner side of turn is almost zero or is exerted as pulling the steering wheel toward the driver. These differences of forces seem to be a cause of the yaw
motion of the upper body. The shoulder support force forms a similar shape as the longitudinal steering force with a larger amplitude and a certain amount of time delay. These areas of the seat back seem to be used for supporting the upper body as well as the steering forces.

4.4 Effect of vehicle motion on lumbosacral joint torque

A further investigation of upper body motion was made by solving inverse dynamics Equation (17), substituting all the measured motions and forces. We focused on the lumbosacral joint that is located between the pelvis and the lumbar (see Figure 18) as we presumed the largest joint torque to be necessary during maneuvers. The lumbosacral joint torque was decomposed into roll, pitch, and yaw components based on the floor coordinate system, shown in Figure 18. In order to examine some of the functions of the decomposed components, another calculation was performed assuming no motion generator movements but the same participant’s movements and external forces.

Figure 19 shows the calculated lumbosacral joint torque in the roll, pitch, and yaw directions. Solid lines in the figure indicate the results with vehicle motion. Dashed lines indicate the results with no vehicle motion. The pitch direction was omitted because almost no difference was seen between the two results.

As is can be seen in the roll component shown in the top graph, the two lines are very different in shape; in particular, the magnitude comes to almost zero in the case of no vehicle motion. This suggests that the roll component of the lumbosacral joint torque might be used to maintain the posture of the upper body against inertial force due to vehicle motion. On the other hand, the difference in the bottom graph is small, and a similar pattern is observed despite there being no vehicle motion. The yaw component of the lumbosacral joint torque seems to be used dominantly for turning the steering wheel.

4.5 Effect of seat back support on lumbosacral joint torque

To investigate the effect of seat back support on lumbosacral joint torque, inverse dynamics were performed by using the measured data for a lane-change maneuver with the following scaled seat back support forces: (a) full support, (b) no side support, and (c) no shoulder support, as shown in Figure 20.

Figure 21 shows the calculated lumbosacral joint torque in the roll and yaw directions. Solid lines in the figure indicate full support case and dashed lines indicate a scaled case. Comparing the two lines, it is found that the effect of side support was observed mainly at the peak of the joint torque,
but a shoulder support effect was seen in particular at a very early phase of turning the steering wheel (around 1.2 sec in the bottom graph of Figure 21). This suggests that the shoulder support force might have more effect on steering wheel operation than the side support force in the low lateral acceleration range.

5 CONCLUSIONS

In this study, we proposed a method for analyzing the interaction between driver and vehicle by combining multibody inverse dynamics and experimentally obtained data. The motion, forces, and internal joint torque of a driver during a slalom and a lane-change maneuver were measured and analyzed to demonstrate the effectiveness of this method.

In the proposed method, a driver in a vehicle is dealt with as a multibody system subjected to various external forces. Combining the motion captured data and measured forces, inertial forces were eliminated and the pressure on the driver was vectorized. Three dimensional measurements of motion and inverse kinematics gave the velocity and angular velocity of any place on the driver’s body, which was visualized by an instantaneous screw axis. Furthermore, it was demonstrated that inverse dynamics with scaling data makes it possible to investigate the driver’s motion from a biomechanical point of view.

The proposed inverse analysis would give significant insights for both biomechanical, neuromus-
cular driver modeling and control theoretic driver modeling. The effects of vehicle motion and seat support on the human internal load will prove helpful in determining vehicle package and seat characteristics in the early phase of vehicle development.

Different driver sizes and driving postures, as well as other maneuvers, need to be further investigated to validate the insights obtained in this study. The contact between the human body and seat needs to consider deformation. In addition to indoor experiments, outdoor measurements are necessary to enhance the scope of application.

REFERENCES


Modelling of sound transfer in middle ear with SMA element

R. Rusinek∗, J. Warminski∗, M. Szymanski#

∗ Department of Applied Mechanics
Faculty of Mechanical Engineering
Lublin University of Technology
Nadbystrzycka 36, 20-618 Lublin, Poland
[r.rusinek, j.warminski]@pollub.pl

# Department of Otolaryngology
Head and Neck Surgery
Medical University of Lublin
Jaczewskiego 8, 20-854 Lublin, Poland
marcinszym@poczta.onet.pl

ABSTRACT

In the paper a three degrees of freedom model of an intact middle ear is presented. The model has a non-linear spring which represents an annular ligament. Dynamics of the model of the intact ear is analyzed near the first and the second natural frequency. Next a model of the reconstructed middle ear with prosthesis made of shape memory alloy (SMA) is proposed where a polynomial dependence describes properties of the shape memory prosthesis. Finally, bifurcation analysis is performed in order to find stapes behavior for various values of excitation.

Keywords: middle ear, shape memory alloy, ossicular prosthesis.

1 INTRODUCTION

The human middle ear consists of three ossicles: a malleus, an incus and a stapes. The ossicles form a sound conduction system which transmits sound from the external ear to the fluids of the inner ear. The ossicles are connected to each other by the incudo-malleal and incudo-stapedial joint. The ossicular chain is supported by two muscles: the tensor tympani muscle attached with its tendon to the handle of the malleus, and the stapedius muscle attached to the stapes neck or posterior crus. The malleus is also firmly connected to the tympanic membrane while stapes is attached to the bony walls of the oval window by annular ligament forming stapediovestibular junction. Such a complex bio-system is modeled in the literature from the last half century. The first study in this field was published in 1961 by M˝oller [1] where also the first scheme of middle ear mechanism was proposed. Next, a similar model was investigated by Zwislocki [2]. In both publications, authors used an electrical circuit to analyse middle ear system. They based on Bárány theory which claims that the ossicles rotate around an axis through the head of the malleus and the short process of the incus. Zwislocki assumed that there is a rigid coupling between malleus and incus, in fact he omitted this joint from the analog circuit. Next stage of middle ear modelling started from 1978 when Finite Element Method (FEM) was used to study spatial vibrations of tympanic membrane and the ossicles [3-10]. Although, FEM is very helpful its accuracy depends on the effort put in geometric and mechanic properties. Moreover, FEM does not allow full description of system dynamics and parameters influence on the system behaviour. Therefore, in the last decades mechanical models are developed where ossicles are represented by lumped masses, connected with springs and dashpots. In the literature, one can find three [11], four [12] and sometimes even six degrees of freedom (dof) models [13]. Generally these models focus only on kinematics of an intact middle ear. Dynamics of these models is analysed hardly ever. Nevertheless, Feng and Gan [13] present the dynamic model described by differential equations of motion in matrix form. They have found the natural frequencies of such a dynamical system. As a result, the displacement of stapes is compared with experimental outcomes giving quite good agreement but still only for intact ossicular chain. Probably, a low number of dynamic middle ear models arises from very complex procedure of obtaining damping and stiffness coefficients which cannot be found in simple experimental tests. The coefficients have to be determined by fitting procedure to get the model response similar to real one. Moreover, in most papers, proposed in the literature, only linear models are considered which cannot explain complexity of middle ear behaviour, specially at higher frequencies.
A middle ear dysfunction is the separate problem which is raised in the medical practice and described in literature. The ossicular chain can be partially destroyed by inflammatory diseases such as chronic suppurative otitis media or cholesteatoma. Then, ossiculoplasty, or reconstruction of the middle-ear ossicles is necessary to improve hearing process of a patient. For more than 50 years middle ear surgery techniques have enabled to improve hearing destroyed by various diseases. Numerous procedures are currently used in clinical practice and a variety of middle ear prostheses are available. The choice of prosthesis type, its material and size is a main problem. One of the main difficulties is adjustment of a prosthesis to a specific anatomical condition in the middle ear. Taking into account a wide range of variables influencing a perfect ossicular reconstruction there is no single ideal ossicular prosthesis. A classical prostheses TORP (Total Ossicular Replacement Prosthesis) and PORP (Partial Ossicular Replacement Prosthesis) are made of titanium. Their length can be adjusted but only once during operation. If the prosthesis is too short, reuse of it is not possible. Our idea consists in developing the smart prosthesis made of modern shape memory alloy (SMA), which would be able to adjust its size and angulation to requirements in a specific medical case. Now, the piston - stapes prostheses (NiTiBOND) made of Nitinol are known in the literature and sold by medical concerns. Clinical tests prove that the acousto-mechanical properties of such self-crimping prostheses are better and less variable because of better sound transmission properties at the incus-prosthesis [14, 15]. However, to authors knowledge there is no study which rises problems of smart PORP or TORPP prostheses made of SMA. Modelling of a shape memory effect (SME) is very difficult for the sake of shape memory and pseudoelastic effect. Generally, SMAs are a family of metals with the ability of changing their shape depending on their temperature [16]. SMAs undergo thermoelastic phase transformations, which can be induced either by temperature or stress. Thermomechanical properties of SMA can be modeled either by microscopic or macroscopic point of view. The microscopic approach treats phenomena in molecular level while from our point of view macroscopic approach is more interesting because takes into account phenomenological features [17].

There is a class of models known as models with assumed phase transformation kinetics which consider preestablished simple mathematical function to describe phase transformation. This approach was first proposed by Tanaka and Nagaki [18]. Next, that work motivated other researchers to present modified transformation kinetics laws Liang and Rogers [19], Brinson [20], among others. These models are probably the most popular in the literature and play an important role in a modelling [17]. Fremond [21] developed a three dimensional model which is able to reproduce the pseudoelastic and shape memory effects by using three internal variables. Very often a one-dimensional model, built up on the original Fremond’s model is developed in many papers [22-26]. Relatively simply one-dimensional model based on Devonshire’s theory was proposed by Falk [27] and Falk and Konopka [28]. This model assumes a polynomial -free energy potential, which allows describe both SME and pseudoelasticity behavior.

There can be one or two-way shape memory effect. The basic difference between one-way and two-way SME is that no reverse change of SMA’s shape occurs in case of one-way SME after subsequent cooling while two-way SME is characterized by a change of SMA’s shape during cooling. For the need of smart prosthesis we use Nitinol element with one-way SME because the prosthesis should change shape after heating and must not recover back when cooling.

Here, a nonlinear three degrees of freedom, but one dimensional model of middle ear is proposed which next is rebuild to a system with ossicular prosthesis made of shape memory alloy (smart prosthesis). The simple one dimensional 3dof model of middle ear cannot reflect complexity of three dimensional ossicles motion, reported for instance in [29-31], but the model is simpler to analyse. Moreover, an introduce of the shape memory element causes difficulties during numerical integrations. That is produced by a strongly nonlinear characteristic of SMA. Therefore, we have decided to use relatively simple model which is able to explain the essence of the smart prosthesis (SP) behavior.
2 DYNAMICS OF INTACT MIDDLE EAR

The human middle ear mechanism can be presented as a 3dof model consisting of three lumped masses connected by seven springs and dashpots (Figure 1). The three ossicular bones: the malleus, the incus, and the stapes are represented by masses $m_m, m_i, m_s$, respectively. The malleus ($m_m$) is jointed with a tympanic membrane (TM) with spring and dashpot $k_{TM}$ and $c_{TM}$. An anterior malleal ligament (AML) suspending the malleus is simulated as spring $k_{AML}$ and dashpot $c_{AML}$. The malleus is connected with the incus by incudomalleal joint (IMJ) represented by the spring $k_{IMJ}$ and dashpot $c_{IMJ}$. Next, the incus and stapes are supported by posterior incudal ligament (PIL) and annular ligament (AL) that are modelled as springs with stiffness $k_{PIL}$ and $k_{AL}$, and dashpots $c_{PIL}$ and $c_{AL}$. The incudostapedial joint (ISJ) is shown as spring $k_{ISJ}$ and dashpot $c_{ISJ}$. Finally, the stapes motion is transferred to cochlea (C) which stiffness and damping is simulated by spring $k_{C}$ and dashpot $c_{C}$. Since stapes motion can be naturally limited by the ear system, the stapes suspension is modeled here as nonlinear. Sometimes, otosclerosis can be presented in the same way as stapes fixation. The spring force of annular ligament $F_{AL}$ has a linear part $k_{AL}x_s$ and nonlinear $k_{1}x_s^2 + k_{2}x_s^3$, then the force of annular ligament is expressed as

$$F_{AL} = k_{AL}x_s + k_{1}x_s^2 + k_{2}x_s^3$$  \hspace{1cm} (1)

Assuming that a motion of the ossicles is stimulated by sound, harmonic excitation $Q\sin \omega t$ is put on the malleus. $Q$ means the amplitude of excitation and $\omega$ the sound frequency. Numerical simulations of the 3dof model of intact ear are performed in Matlab-Simulink using 4-th order Runge-Kutta method with automatic steps control. Middle ear parameters are taken from [32] and presented in Table 1. The stiffness coefficients give the first and the second natural vibrations frequency of 3dof system convergent with the experimental results reported in [33].

Results of the numerical analysis of the two resonances obtained for the excitation $Q = 0.12$ mN, are depicted in Figure 2, where $A_M$, $A_I$ and $A_S$ means the amplitude of the malleus, the incus and the stapes, respectively. The dashed line represents unstable solutions whereas, the solid line depicts stable ones.

The stapes amplitude ($A_s$) at the first resonance (Figure 2a) is smaller then the malleus ($A_m$) and incus ($A_I$) amplitude that is exactly inversely to the second resonance (Figure 2b). The dashed line
### Table 1. Parameters of 2dof model of intact middle ear.

<table>
<thead>
<tr>
<th>mass</th>
<th>[mg]</th>
<th>stiffness [N/mm]</th>
<th>damping [Ns/m]</th>
</tr>
</thead>
<tbody>
<tr>
<td>$m_M$</td>
<td>25</td>
<td>$k_{TM}$</td>
<td>0.3</td>
</tr>
<tr>
<td>$m_I$</td>
<td>28</td>
<td>$k_{AML}$</td>
<td>0.8</td>
</tr>
<tr>
<td>$m_S$</td>
<td>1.78</td>
<td>$k_{IMJ}$</td>
<td>1000</td>
</tr>
<tr>
<td>$k_{ISJ}$</td>
<td>1.35</td>
<td>$c_{ISJ}$</td>
<td>0.000389711</td>
</tr>
<tr>
<td>$k_{PIL}$</td>
<td>0.4</td>
<td>$c_{PIL}$</td>
<td>0.000981495</td>
</tr>
<tr>
<td>$k_{AL}$</td>
<td>0.623</td>
<td>$c_{AL}$</td>
<td>0.0000328535</td>
</tr>
<tr>
<td>$k_{1AL}$</td>
<td>300 N/mm²</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$k_{2AL}$</td>
<td>$2.25 \times 10^6$ N/mm²</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$k_C$</td>
<td>0.2</td>
<td>$c_C$</td>
<td>0.0000525655</td>
</tr>
</tbody>
</table>

**Figure 2.** The first (a) and the second (b) resonance of 3dof model of intact middle ear

represents the unstable solution which cannot be achieved practically. Now, the attention is paid on the stapes dynamics because its motion is directly transferred into an inner ear. When the excitation frequency $f$ increases, the amplitude jumps down from point B to C, whereas while decreasing frequency the amplitude jumps up from point D to E (Figure 2). This jump phenomenon exists both for the first and the second resonance. Moreover, an interesting discontinuities are noticeable near the first resonance. To explain what happens here, a bifurcation diagram of stapes displacement ($x_s$) is presented in Figure 3. The red colour represents classical stroboscopic map (Poincaré sections) where the system output is collected once at the excitation period $T = 2\pi/\omega$. Since, the system output is suspected of subharmonic vibrations additionally, the black colour means the points of phase trajectory where stapes velocity equals zero ($v_s = 0$). Starting from $f = 0.98$ kHz indeed subharmonic motion exists with the jump of vibrations amplitude at $f = 1$kHz, $f = 1.02$kHz and $f = 1.1$kHz. The motion of period $1T$ and subharmonic oscillations are depicted in phase diagrams in Figure 4a and 4b, respectively.

### 3 EAR MODEL WITH SMA PROSTHESIS

Sometimes middle ear disfunction, such as otosclerosis or ossicular chain damaged must be treated with the help of ossicular prostheses. Here, a new type of the smart PORP (SPORP), which is placed instead of a malleus and incus is proposed. In this case the 3dof model of intact ear is reduced to 2dof system presented in Figure 5. The smart prothesis (SPORP) is made of nitinol
Figure 3. Bifurcation diagram of intact middle ear near the first resonance

Figure 4. Phase space of system with harmonic (a) and subharmonic (b) motion

as a rod of diameter $d$, length $L$ and cross section $A$. The prosthesis is schematically presented in Figure 5 as a spring. Then, a restoring spring force $F_{SMA}$ is given as

$$F_{SMA} = \sigma A \tag{3}$$

Stress $\sigma$ in SM (shape memory) element is described using the polynomial model proposed by Falk [27]. The model is based on Devonshire’s theory and considers a polynomial-free energy. It was initially proposed for a one-dimensional media and later extended for three-dimensional context [28]. According to this model, neither internal variables nor dissipation potential is necessary to describe pseudoelasticity and SME [17]. Therefore, the only state variables for this model are strain $\varepsilon$ and temperature $T$. Then, the free energy is chosen in such a way that the minima and maxima points represent stability and instability of each phase of the SMA. As it is reported in [17], usually, in one-dimensional models three phases are considered: austenite (A) and two variants of martensite (M+, M-). Hence, the free energy is chosen such that for high temperatures ($T > T_A$), it has only one minimum at vanishing strain, representing the equilibrium of the austenitic phase. At low temperatures ($T < T_M$), martensite is stable, and the free energy must have two minima at nonvanishing strains. At intermediate temperatures ($T_M < T < T_A$), the free energy must have equilibrium points corresponding to both phases. Therefore, the free energy should be defined
as a sixth-order polynomial equation in such way that the minima and maxima points represent stability and instability of each phase of the SMA. Thus, the following free energy potential is proposed

$$W(\varepsilon, T) = \frac{a}{2}(T - T_M)\varepsilon^2 - \frac{b}{4}\varepsilon^4 + \frac{b^2}{24a(T_A - T_M)}\varepsilon^6$$  \hspace{0.5cm} (4)$$

where $a$ and $b$ are positive material constants. Finally, the polynomial model of SMA is given by

$$\sigma = \frac{\partial W}{\partial \varepsilon} = a(T - T_M)\varepsilon - b\varepsilon^3 + \frac{b^2}{4a(T_A - T_M)}\varepsilon^5$$  \hspace{0.5cm} (5)$$

The SM spring force according to Figure 5 depends on displacements $x_P$ and $x_S$. Taking into consideration Equation 3 is defined as

$$F_{SMA} = \frac{aA}{L}(T - T_M)(x_P - x_S) - \frac{bA}{L^3}(x_P - x_S)^3 + \frac{Ab^2}{4aL^5(T_A - T_M)}(x_P - x_S)^5$$  \hspace{0.5cm} (6)$$

Substituting

$$a_1 = \frac{aA}{L}, \hspace{0.5cm} a_2 = \frac{bA}{L^3}, \hspace{0.5cm} a_3 = \frac{Ab^2}{4aL^5(T_A - T_M)}$$  \hspace{0.5cm} (7)$$

the SM spring force takes the shorter form

$$F_{SMA} = a_1(T - T_M)(x_P - x_S) - a_2(x_P - x_S)^3 + a_3(x_P - x_S)^5$$  \hspace{0.5cm} (8)$$

Thus, the differential equations of motion of 2dof system with SM element is given as follows

$$m_P\ddot{x}_P + k_{TM}x_P + F_{SMA} + c_{TM}\dot{x}_M = Q\sin \omega \tau$$  \hspace{0.5cm} (9)$$

$$m_S\ddot{x}_S - F_{SMA} + k_{CS}x_S + k_{ALS}x_S + k_{2ALS}x_S^2 + k_{3ALS}x_S^3 + c_{CS}\dot{x}_S + c_{ALS}\dot{x}_S = 0$$

Since, diameter ($d$) and length ($L$) of the rod influence on stiffness ($F_{SMA}$ force) and mass of the prosthesis, the prosthesis mass ($m_P$) is written as follows

$$m_P = \frac{\pi d^2}{4}L\rho$$  \hspace{0.5cm} (10)$$

where, $\rho$ means nitinol density. Prosthesis mass should be matched precisely so that natural vibrations frequency of the reconstructed ear is close by frequency of intact ear. Therefore, in the next subsection an analysis of simple linear case is done in order to find the proper rod diameter ($d$).

3.1 Linear vibrations

In case of linear 2dof system $a_2 = a_3 = k_{1ALS} = k_{1ALS} = 0$. Then, the differential equation of motion can be presented in matrix form

$$\ddot{X} + KX = 0$$  \hspace{0.5cm} (11)$$
where

\[
\begin{align*}
\ddot{X} &= \begin{bmatrix} \ddot{x}_p \\ \ddot{x}_s \end{bmatrix}, \quad X = \begin{bmatrix} x_p \\ x_s \end{bmatrix}, \quad M = \begin{bmatrix} m_p & 0 \\ 0 & m_s \end{bmatrix}
\end{align*}
\]

(12)

\[
K = \begin{bmatrix}
k_{TM} + a_1 T - a_1 T_M & a_1 T_M - a_1 T \\
a_1 T_M - a_1 T & a_1 T - a_1 T_M + k_C + k_{AL}
\end{bmatrix}
\]

(13)

Taking the model parameters from Table 1 and SMA properties from Table 2 [17], the eigenvalue problem is solved and natural frequencies of the system are calculated as a function of rod diameter \(d\) and shown in Figure 6.

\begin{table}[h]
\centering
\begin{tabular}{|c|c|}
\hline
\textbf{d} & \textbf{1 mm} \\
\hline
\textbf{\(a\)} & 1000 MPa/K \\
\hline
\textbf{\(b\)} & \(40 \times 10^6\) MPa/K \\
\hline
\textbf{T} & 309.6 K \\
\hline
\textbf{T_M} & 287 K \\
\hline
\textbf{T_A} & 313 K \\
\hline
\textbf{L} & 5 mm \\
\hline
\textbf{\(\rho\)} & 6500 kg/m³ \\
\hline
\end{tabular}
\caption{Parameters of shape memory element.}
\end{table}

Figure 6. The first (a) and the second (b) eigenfrequencies of 2dof model versus rod diameter.

The first natural frequency of 2dof system decreases with the rod diameter \(d\). For practical reason it is assumed that \(d=1\) mm because then, the first eigenfrequency \(f_1 \approx 1\) kHz. That is similar with the first eigenfrequency of 3dof model of intact ear and first of all with experimental tests. The second eigenfrequency \((f_2)\) is much further then the threshold of audibility, so that is out of our consideration.

3.2 Non-Linear vibrations

The nonlinear 3dof model of reconstructed middle ear with the help of smart prostheses is tested here. Parameters of shape memory element (prosthesis) and 2 dof model are taken from Table 2 and Table 1, respectively. The Runge-Kutta algorithm of the 4-th order, implemented in Matlab, is used to perform numerical analysis of Equations (9). It is assumed that temperature of the prosthesis is 36.6 Celsius degrees (309.6K), the excitation \(Q = 4.8\) N and bifurcation analysis of the stapes motion performed (Figure 7).
Frequency of external excitation is set as a bifurcation parameters. Interestingly, for the assumed parameters the system demonstrate only irregular motion (Figure 8) even for $f = 1020$Hz where the linear model has the first natural frequency. Although, this behavior is interesting from non-linear mechanics point of view, but practically application of the smart prosthesis is questionable. Therefore, an additional viscous damping is introduced into SM element. It is assumed that the damping coefficient $c_{SMA}$ is referred to $c_{IMJ}$. The bifurcation diagram showing an influence of the damping coefficient $c_{SMA}$ on stapes dynamics is presented in Figure 9. Figure 9a shows classical stroboscopic Poincaré map, which suggests harmonic motion for $c_{SMA}/c_{IMJ} > 80$ while the system output, presented as a trajectories in phase space (Figure 10), demonstrate sub-harmonic vibrations. Similarly the range of $c_{SMA}/c_{IMJ}$ between 25 and 35 also is suspected on regularity, meanwhile only sub-harmonic motion is revealed as well. Less sub-harmonics exist for stronger damping but the stapes answer is still far for harmonic motion. In case of practical application of smart prosthesis it is expected to have regular vibrations.
Figure 9. Bifurcation diagrams of 2dof system with additional damping. Stapes displacement versus excitation frequency.

Figure 10. Phase space of 2dof model for $c_{SMA}/c_{IMJ} = 30$ (a) and $c_{SMA}/c_{IMJ} = 100$ (b).

4 CONCLUSIONS

The three degrees of freedom model of middle ear with nonlinearities demonstrates interesting behaviour in the range of assumed parameters near the first resonance. The sub-harmonic system response with the amplitude jumps is caused by annular ligament nonlinearity. From practical point of view, the presented results point that in case of stapes fixation e.g due to otosclerosis, different kinds of solutions can be obtained. Probably, also other harmonic outcomes are possible depending on initial conditions. This problem is not presented here, but will be developed.

The second part of the work is devoted to reconstructed middle ear with the help of shape memory prosthesis. This method of treatment is innovative and requires more detailed analysed. For the sake of strong nonlinearity of SMA characteristic the answer of middle ear structure is complex despite the fact the model has only two degrees of freedom. Therefore, a choice of smart prosthesis diameter and damping properties of the prosthesis material have essential meaning. Authors believe that among various possible system responses one can find interesting ones from scientific and also practical point of view. More results will be discussed in an extended version of the paper, which will be devoted to full parametric analysis of SM prosthesis.
Acknowledgements

The work is financially supported under the project of National Science Centre (Poland) no. 2014/13/B/ST8/04047.

REFERENCES


ON THE EFFECT OF ATTACHMENT POSITION AND COMPLIANCE OF WEARABLE ROBOTS ON HUMAN JOINT AND INTERFACE FORCES

Jochen Damerau, Jovana Jovic, Takao Watanabe, Udo Wolz

Bosch Corporation
3-6-7 Shibuya, Shibuya-ku, Tokyo 158-0086, Japan
[jochen.damerau, jovana.jovic, takao.watanabe,
udo.wolz]@jp.bosch.com

ABSTRACT

In the past decades we witnessed the increased research efforts in the field of wearable robotic devices. The development of exoskeletons still faces many challenges and constraints related to the close interaction to a human operator. The influence of exoskeleton assistance to the human body is still not well studied and understood. In the literature several authors proposed dynamics models of exoskeletons attached to the human body. However, no study has been performed in order to investigate the influence of attachment position and stiffness properties on whole body human dynamics.

In this study we investigated the effect of the positions and elastic properties of the exoskeleton attachments to human lower limbs joint force and torque values and the forces that occur at the interface between human and robotic device during the execution of the squat exercise in the sagittal plane. We used 3 degrees of freedom (DOF) inverted pendulum model to model human body and 2-DOF model to model exoskeleton dynamics. The values of exoskeleton attachments stiffness properties and locations relatively to the human body were varied in order to see their influence on human body dynamics. We showed that wearing exoskeleton device presented in this study would reduce lower limb joint torque values. Moreover, we showed that both attachments stiffness values and positions influence the human joint torque and human-exoskeleton interaction forces.

In the future we plan to investigate human-machine interaction (HMI) during frequently executed daily activities such as walking, sit-to-stand and stairs ascending/descending by conducting dynamic simulations using more detailed models based on commercial software. The results will be experimentally validated.

Keywords: Exoskeleton, attachment positions, attachment stiffness, human dynamics and kinematics.

1 INTRODUCTION

The first human exoskeleton devices were designed in the middle of the last century in two countries in parallel, in Serbia, the former Yugoslavia at the time [1]-[3], and in the United States [4]-[7]. The research done in the former Yugoslavia was focused on developing assistive technologies for physically disabled persons, such as paraplegic and hemiplegic patients [3]. While the research done in the United States mainly focused on developing exoskeletons for military use which would improved the physical performances of able-bodied persons [4]-[7].

Today, about fifty years later, the wearable exoskeletons are continued to be developed for different purposes targeting primarily motion restoration for persons with paraplegia due to spinal cord injury (SCI), motion rehabilitation and training for stroke survivals and improvement of the physical performances of able bodied persons, such as soldiers, disaster workers and elderly subjects.

Wearable robotic exoskeletons designed for the motion restoration of persons with paraplegia due to SCI enable the users to stand up and walk again and offer them some of the health benefits associated with an upright posture such as reduce of the muscle atrophy, pressure sores
appearance, and the risk of the bladder infections, improving blood circulation, prevention of loss of bone density, muscle mass and changes in body fat distribution [8]-[9]. In addition to the therapeutic benefits, the use of the exoskeletons would allow paraplegics to reach further than when sitting in a wheelchair and to communicate with other people on an equal level. Today, a few robotic exoskeletons for motion restoration of paraplegic individuals are commercially available. Some of them are ReWalk from Argo Medical Technologies [10] the eLegs from Berkley Bionics [11], and Hybrid Assistive Limb (HAL) from Cyberdine [12]. Those devices have powered hip and knee joints. ReWalk and eLegs require that the user provides balance with the aid of crutches and uses torso motions, arm movements, and/or a push button interface. A new version of commercially available medical HAL device, HAL-5 LB (Type C) uses electromyographic signals of the user’s muscles to detect motion initiative to generate the movement. In addition to commercially available exoskeletons for motion restoration of subjects with SCI, a few exoskeletons with the same purpose have been developed in research facilities worldwide. A detailed lower limb exoskeletons review can be found in [13].

Beside the motion restoration, a number of robotic exoskeleton devices have been developed for gait rehabilitation of stroke survivors. They enable long training sessions, high accuracy of desired motion patterns and a reduced workload for physical therapists. Some of the examples of rehabilitation exoskeletons are LOwer-extremity Powered ExoSkeleton (LOPES) [14], Lokomat [15], and Active Leg Exoskeleton (ALEX) [16].

In the past decades we also witnessed the increased research efforts in the field of wearable robotics for augmentation of physical performances of able bodied subjects, such as BLEEX [17], ExoClimber and ExoHiker [18]. Those exoskeletal devices allow the user to carry a heavy backpack while walking long distances [18], performing squat, bend and twist motions [17], and walk on ascending and descending slopes [18]. A few robotic devices, such as commercially available Walking Assist Pelvic Support exoskeleton designed by Honda [19] and the EXPOS designed at Sogang University [20], were developed as a walking assist device for elderly subjects.

Despite the differences in the intended use, development of exoskeletons faces the same challenges and constraints, particularly those related to interfacing closely to a human operator. The influence of exoskeleton assistance to the human body is still not well studied and understood. A few studies have been done in order to model biomechanical interaction between human upper limbs and rehabilitation robots [21] and between human upper limbs and exoskeletons [22]-[24]. In [21] - [23] authors used commercially available software [25], while in [24] authors developed a simplified model of human upper extremities. In [26] the whole body HMI has been studied under two different attachments constrains using AnyBody biomechanical modelling software [25]. In above mentioned studies a few approaches have been proposed to model the physical HMI. Some sources (see [27], [22]-[23]) suggest using kinematical constraints while in [26] a unilateral force element and in [24] a spring-damper is used for modelling the strap-type attachments. However, according to our knowledge, no study has been performed in order to systematically investigate the influence of attachment position and stiffness properties on whole body human dynamics.

In this qualitative study we investigated the effect of the positions and elastic properties of the exoskeleton attachments to human lower limbs on the quantities that are relevant for evaluation of performances and comfort of wearable robots during the execution of a periodic squat exercise in the sagittal plane. These are in particular the force and torque values in the human joints and the forces that occur at the interface between human and device. In this study we chose to study HMI dynamics during the execution of a periodic squat exercise because of the usage of the exercise in the fields of rehabilitation and strength and conditioning [28]-[29]. Moreover, the squat exercise could be seen as a prerequisite for more complex tasks of daily activities such as picking up an item, descending the stairs, or rising from a chair [30].

The rest of the article is organized as follows: Section 2 explains the biomechanical model of a human body and a wearable exoskeleton device, as well as the modelling strategy used to
simulate the HMI during execution of the squat exercise. Results are presented and discussed in Sections 3. Concluding remarks and perspectives for future work are given in Section 4.

2 METHOD

2.1 Biomechanical models of the human body

The biomechanical model of the human body used in this study is shown in Fig. 1. We analyzed a periodic squat exercise in the sagittal plane. The human body was modeled using 3 DOF inverted pendulum model composed of three rigid segments (shanks \( l_1 \), thighs \( l_2 \), and head arm trunk (HAT) segment \( l_3 \)) connected by revolute joints (see Fig. 1). We supposed a perfect sagittal symmetry during the execution of the squat exercise. In this study we used the Winter anthropomorphic database [31] to estimate segmental and inertial parameters of the human body. The simulated human subject was male gender, with body height of 1.75 m and body weight of 70 kg. We supposed that the human model was rigidly attached to the ground.

It has been reported [32] that the presence of the exoskeleton support would modify human joint kinematics. Hence, in order to allow the exoskeleton to influence human joint trajectories, we modeled human motor control during execution of the squat exercise using a PD controller in joint space. Accordingly, the human joint torque values due to the periodic motions of the squat exercise could be calculated as follows:

\[
T_{\text{human}} = K_p (Q_{\text{ref}} - Q) + K_d (\dot{Q}_{\text{ref}} - \dot{Q}) \tag{1}
\]

In Equation (1) \( T_{\text{human}} \) represents the vector of joint torque values. Vectors \( Q_{\text{ref}}, \dot{Q}_{\text{ref}} \), and \( Q \) contain desired and achieved joint trajectories and their derivates for the hip, knee and ankle joints, respectively. \( K_p \) and \( K_d \) represents the proportional and derivative gains, respectively. The desired joint trajectories were expressed using cosine function:

\[
Q_{i\text{ref}} = a_i \cos(\omega t) - a_i \tag{2}
\]

for \( i=1,2,3 \). Where \( t \) represents the time vector, \( \omega \) is angular frequency, and \( Q_{1\text{ref}}, Q_{2\text{ref}}, \) and \( Q_{3\text{ref}} \) represent ankle, knee and hip desired joint trajectories, respectively. Constants \( a_1, a_2, \) and \( a_3 \) had values of 0.22, 0.6, and 0.8, respectively.

2.2 Mechanical models of the exoskeleton device

The 2-DOF mechanical model of the exoskeleton device used in this study is shown in Fig. 1. The exoskeleton model was composed of two rigid segments with the length of \( l_4 \) and \( l_5 \) connected by revolute joints, \( q_4 \) and \( q_5 \). Lengths \( l_4 \) and \( l_5 \) had the same values as the lengths of the human shank \( l_1 \), and human thigh \( l_2 \), respectively. The exoskeleton model was rigidly attached to the ground. In this study we supposed that the exoskeleton has perfect position control of joint angles. Hence the joint angles of exoskeleton had predefined values expressed by equation (2). The human and exoskeleton models are connected by spring-elements with stiffness values of \( C_i \) at the interface locations \( d_i \) (\( i=1, 2, 3 \)) shown in Figure 2.
2.3 Modelling of HMI

The schematic representation of the human exoskeleton interaction modelling is given in Figure 3. Knowing the vectors of human and exoskeleton joint angle, the positions of the human and exoskeleton joints could be expressed in Cartesian space using the corresponding Forward Kinematic Models (FKM). Further, the vector of interaction forces between human and exoskeleton device, $F$, could be calculated as follows:

$$ F = C(P_{human} - P_{exoskeleton}) $$  \hspace{1cm} (3)

where $P_{human}$ represents vector of Cartesian positions of human hip, knee, and ankle joints, and $P_{exoskeleton}$ represents vector of Cartesian positions of exoskeleton joints. Stiffness values of the exoskeleton attachments at the three attachment positions are given with the matrix $C$.

Further the vector of disturbances in human joint torque values due the human-exoskeleton interaction, $T_{exoskeleton}$, could be calculated as follows:

$$ T_{exoskeleton} = J^T F $$  \hspace{1cm} (4)

where $J^T$ represent Jacobian matrix at the attachment positions that map human-exoskeleton interaction forces to the human joint space.

Finally, combining Equation (1) and Equation (4), the vector containing the total values of human joints torque, $T$, could be computed as:
Knowing the vector $T$, the achieved joint trajectories of the human subject wearing the exoskeleton device during the squat exercise could be computed using Forward Dynamic Model (FDM) of rigid body structures [33]:

$$H(Q)\ddot{Q} + C(Q, \dot{Q})\dot{Q} + G(Q) = T$$

where $H(Q)$ is inertia matrix, $C(Q, \dot{Q})$ is the vector of Coriolis and centrifugal forces, and $G(Q)$ is the gravitational term.

### 2.4 The goal of the study

The goal of this study was to perform qualitative investigation which would give us an insight into the influence of the attachment properties, positions $d_i$ and stiffness represented by the matrix $C_i$ ($i=1,2,3$), on force and torque values in the human joints and the forces that occur at the interface between human and exoskeleton device. Matrix $C_i$ consists of identical entries $c_i$ on the main diagonal of the matrix and zero elements otherwise. Consequently, we performed simulations for two different locations of exoskeleton attachments. In the first simulation condition we positioned the exoskeleton attachments at the lowest quarter of the shank segment, at the lowest quarter of the thigh segment and at the hip joints level. In the second simulation condition we positioned the exoskeleton attachments at the middle of the human shank segment, at the middle of the human thigh segment and at the hip joints level. Locations of exoskeleton attachments for two different simulation conditions are shown in Fig. 2. In the both simulation conditions we varied each element of the stiffness vector to values of 100 N/m, 1000 N/m, and 3000 N/m. The simulation time was set to 10 s with the sampling frequency of 1000 samples per second. The simulations were performed using custom software designed in MATLAB programming language.

![Figure 2. Exoskeleton attachment positions $d_i$ for $i=1,2,3$ for two simulation conditions. $c_i$ represents the stiffness value of the attachment at the locations $d_i$. Values of attachment positions in the first simulation conditions were following: $d_1=0.25*l$, $d_2=0.25*l$. Values of attachment positions in the second simulation conditions were following: $d_1=0.5*l$, $d_2=0.5*l$.](image-url)
3 RESULTS

3.1 Influence of attachment properties on human joint force and torque values

In order to estimate the effect of the attachment positions and stiffness properties on human body dynamics, for each simulation case we computed the mean value over the time of the absolute resultant force and torque at human ankle, knee, and hip joints. The results are shown in Fig. 4, Fig. 5, Fig. 6 and Fig. 7. The mean values of the absolute resultant force and torque at hip, knee and ankle joints calculated over squat exercise simulated subject performed without wearing exoskeleton device is also dispatched in Fig. 4 - Fig. 7.

From the results dispatched in Fig. 4, and Fig. 5, it can be observed that for both simulation conditions the increase of stiffness values of the exoskeleton attachments slightly decreases resultant forces at human joints. Contrarily, increasing the stiffness of the exoskeleton attachments notably decreases human joint torque values (see Fig. 6 and Fig. 7). Moreover, lower limbs joint torque values computed during second simulation condition are slightly lower compared with joint torque computed during the first simulation condition. As expected, that variation of stiffness and position of the exoskeleton attachments do not influence torque values at the human hip joint.
Figure 4. The mean values of the resultant hip, knee, and ankle joint force over the first simulation condition. For more information about the exoskeleton attachment positions during simulation conditions see Fig. 2.

Figure 5. The mean values of the resultant hip, knee, and ankle joint force over the second simulation condition. For more information about the exoskeleton attachment positions during simulation conditions see Fig. 2.
Figure 6. The mean values of the absolute hip, knee, and ankle joint torque over the first simulation condition. For more information about the exoskeleton attachment positions during simulation conditions see Fig. 2.

Figure 7. The mean values of the absolute hip, knee, and ankle joint torque over the second simulation condition. For more information about the exoskeleton attachment positions during simulation conditions see Fig. 2.
Figure 8. The mean values of the human-exoskeleton interaction forces in horizontal and vertical direction over the first simulation condition. For more information about the exoskeleton attachment positions during simulation conditions see Fig. 2.

Figure 9. The mean values of the human-exoskeleton interaction forces in horizontal and vertical direction over the second simulation condition. For more information about the exoskeleton attachment positions during simulation conditions see Fig. 2.
3.2 Influence of attachment properties on human-exoskeleton interaction forces

We computed the mean value of the human-exoskeleton interaction forces in horizontal and vertical direction over the time. The results are presented in Figure 8 and Figure 9. It can be noticed that the human-exoskeleton interaction forces have higher values in the second simulation case.

CONCLUSIONS

In this study we used a 2-DOF model of the lower limb exoskeleton device and a 3-DOF model of the human body to study the influence of the exoskeleton attachment positions and stiffness values on the human joint force and torque values and the forces that occur at the interface between human and exoskeleton during the execution of a squat exercise. We have shown that wearing the lower limb exoskeletal device presented in this study could reduce knee and ankle joint torque during the squat exercise. Both stiffness values and the position of attachments between robotic device and human body play an important role in human joint torque reduction. Increasing the stiffness values of the exoskeleton attachments reduces the human lower limbs joint torque. Placing the exoskeleton attachments in the middle of the human thigh and shank segments would have stronger influence on the reduction of the human lower limbs joint torque. At the same time, increase of attachments stiffness values increases the interaction forces between human and exoskeleton device. In addition, the human-exoskeleton interaction forces have higher values in the case when the exoskeleton attachments are located in the middle of the human thigh and shank segments. Therefore, in order to design an effective, safe and comfortable wearable robotic device a good trade-off between human-exoskeleton attachments properties must be found. We also had shown that the chosen exoskeleton has low influence on the resultant forces at the human joints.

In the future, we plan to conduct dynamic simulations in order to investigate HMI during frequently executed daily activities such as walking, sit-to-stand and stairs ascending and descending using more detailed models based on commercial software such as AnyBody [25] or OpenSim [34]. The predicted HMI force will be experimentally evaluated. The optimal human-exoskeleton attachment positions and stiffness properties which induce significant joint force and torque reductions and at the same time try to minimize interaction forces between the human body and wearable robotic device will be investigated using dynamic optimization approach.

REFERENCES


[34] National Center for Simulation in Rehabilitation Research, NCSRR, CA, USA. Available: http://opensim.stanford.edu/
The Influence of Neuromusculoskeletal Model Calibration Method on Predicted Knee Contact Forces during Walking

Gil Serrancoli*, Allison L. Kinney#, Benjamin J. Fregly†, Josep M. Font-Llagunes*

* Department of Mechanical Engineering
Universitat Politècnica de Catalunya
Av. Diagonal 647, 08028 Barcelona, Catalunya
gil.serrancoli@upc.edu, josep.m.font@upc.edu

# Department of Mechanical and Aerospace Engineering
University of Dayton
300 College Park, OH 45469, Dayton, USA
akinney2@udayton.edu

† Department of Mechanical and Aerospace Engineering
University of Florida
231 MAE-A Building, FL 32611-6250, Gainesville, USA
fregly@ufl.edu

ABSTRACT

Neuromusculoskeletal models used to predict muscle and joint contact forces for a specific individual require specification of muscle-tendon, skeletal geometry, and neural control model parameter values. Though these parameter values should ideally be calibrated using in vivo data collected from the subject, they are often taken from generic models. This study explored the influence of three model calibration methods on predicted lower limb muscle and knee contact forces during walking. The calibrated model from each approach was used in a static optimization that predicted knee contact forces for six walking trials. The predictions were evaluated using knee contact forces measured in vivo from a subject implanted with a force-measuring knee replacement. The first calibration approach used muscle-tendon model parameter values (i.e., optimal muscle fiber lengths and tendon slack lengths) taken directly from the literature. The second approach calibrated muscle-tendon model parameter values such that each muscle operated within a physiological range on the ascending region of its normalized force-length curve. The third approach used a novel two-level optimization that exploited knowledge of the knee contact force measurements to calibrate muscle-tendon, moment arm, and neural control model parameter values such that the calibrated model would predict the in vivo contact forces as closely as possible. For the third approach, three walking trials were used to calibrate the model and the remaining three to test the calibrated model. Overall, calibration method had a large affect on predicted knee contact forces. The first method produced highly inaccurate contact force predictions and infeasible solutions for most time frames. The second approach produced accurate medial contact force predictions (average $R^2 = 0.89$, average RMS error = 107 N) but inaccurate lateral predictions (average $R^2 = -1.77$, average RMS error = 297 N). The third approach produced accurate testing predictions for both medial (average $R^2 = 0.91$, average RMS error = 96 N) and lateral (average $R^2 = 0.76$, average RMS error = 84 N) contact force. These results reveal that when knee contact force data are available, a single set of model parameter values can be successfully calibrated to predict medial and lateral knee contact force accurately over multiple walking cycles. They also reveal that when knee contact force data are not available (the most common situation), a simple calibration method based on muscle operating ranges on their normalized force-length curves may be sufficient to produce accurate medial but not lateral knee contact force predictions.

Keywords: Knee contact forces, muscle force estimation, musculoskeletal model calibration, static optimization, biomechanics

1 INTRODUCTION

The ability to determine muscle and joint contact forces accurately during human movement could be useful for various medical applications, such as evaluation of injured subjects at follow-up or prediction of surgical outcome in advance. Experimental measurement of muscle and joint contact forces is not practical in a clinical setting and currently would require invasive
measurement methods (e.g., placing buckle force transducers on tendons). For this reason, numerical methods have been proposed as an alternate means for determining these forces. However, there is indeterminacy in the muscle force calculation process, since the human musculoskeletal system possesses many more muscles than degrees of freedom. Consequently, optimization methods are often applied to solve the indeterminacy problem.

The most common optimization approaches found in the literature are static and dynamic optimization [1,2]. Both are based on the idea that the central nervous system follows a strategy that minimizes some physiological variable (cost function) subject to various constraints. When available, in vivo hip or knee contact force measurements can be used to evaluate lower limb muscle force predictions, although such an approach does not guarantee that the predicted muscle forces will be accurate. Several studies have followed such an approach [3–5]. However, no study has been able to calibrate muscle-tendon, moment arm, and neural control parameter values in a lower limb neuromusculoskeletal model such that the model can predict medial and lateral knee contact forces accurately for multiple walking trials not used in the calibration process. Furthermore, most studies use neuromusculoskeletal model parameter values taken directly from the literature rather than calibrated to the unique functional characteristics of the subject being modeled.

To calibrate parameter values in a neuromusculoskeletal model, researchers should use as much available experimental data as possible to constrain the calibration process. When information is missing, model parameter values should be constrained to remain within physiologically realistic bounds whenever possible. Muscle activations can also be constrained using experimental muscle synergy information [6], potentially reducing the amount of indeterminacy in the muscle force calculation process [5].

The goal of this study was to investigate how predicted leg muscle and knee contact forces differ for three model calibration approaches. Each approach used static optimization applied to a subject-specific musculoskeletal model to estimate muscle forces for six normal walking trials collected from a subject implanted with a force-measuring knee replacement. The first approach used a standard method where all muscle-tendon model parameter values were taken directly from the literature without adjustment or scaling. The second approach pre-calibrated all muscle-tendon model parameter values such that each muscle operated within a physiologically reasonable range on its normalized force-length curve [7,8]. The third approach used a synergy-based two-level optimization formulation that calibrated muscle-tendon as well as moment arm and neural control model parameter values such that static optimization reproduced experimental knee contact force measurements. For all three approaches, three walking trials were used for model calibration and three for testing knee contact force predictions generated by the calibrated model. The results highlight the significant impact that poorly calibrated neuromusculoskeletal model parameter values can have on predicted knee contact and leg muscle forces.

2 METHODS

2.1 Experimental data

Experimental data were taken from the Fourth Grand Challenge Data Competition to Predict In Vivo Knee Loads [4]. Kinematics (marker trajectories and knee fluoroscopy), ground reaction forces/torques, and electromyographic (EMG) data were used from six overground gait cycles (self-selected speed: 1.26 ± 0.03 m/s) of a subject (gender: male, age: 88 years, mass: 65 kg, height: 166 cm) implanted with an instrumented tibial tray. In vivo knee contact force measurements were available for the medial and lateral sides [4]. EMG data were measured for ten lower limb muscles (Adductor Magnus - Addmag; Biceps Femoris Long Head - Bflh; Gastrocnemius Lateralis - GasLat; Gastrocnemius Medialis - GasMed; Peroneus Longus - PerLong; Semimembranosus - Semimem; Soleus - Sol; Tibialis Anterior - TibAnt; Tensor Fascia Latae - TFL; Vastus Lateralis – VasLat). These data were high-pass filtered (fourth-order zero-phase-lag Butterworth filter at 30Hz), rectified, low-pass filtered (fourth-order zero-phase-
lag Butterword filter at 6 Hz) and normalized by the maximum values of all available movement trials. For consistency, knee contact and ground reaction forces were also low-pass filtered (fourth-order zero-phase-lag Butterworth filter at 6 Hz).

2.2 Muscle synergy analysis

Experimental muscle activations were calculated for all six gait trials using an activation dynamics model [9,10]. From these data, a muscle synergy analysis was performed to decompose the activation signals into time-varying neural commands (NCs) (separate for each trial), which represent low-dimensional activation patterns, and corresponding synergy vectors (SVs) (common for all trials), which contain weights defining how each NC contributes to the activation of each muscle [6,11]. A non-negative matrix factorization approach was used to decompose the signals [12]. Muscle synergy information was used in the third calibration approach in an attempt to decrease the amount of indeterminacy in the muscle force calculation process (Section 2.4). To select the number of NCs and SVs (modules) used to parameterize muscle activations for the third calibration approach, we picked the minimum number of modules required to reconstruct activation signals with a variance accounted for higher than 90%, which was five.

2.3 Inverse kinematics and dynamics analyses

A patient-specific musculoskeletal model developed in OpenSim [13] was used to calculate inverse kinematics and dynamics results. The bone geometry of the model was obtained from a CT scan of the subject being modeled [4], while muscle origin and insertion points were defined by scaling a published OpenSim model [14] and then projecting the points to the nearest locations on the subject-specific bone models. The model consisted of the pelvis and the right leg (femur, patella, tibia/fibula, and foot) and possessed 24 degrees of freedom (DOF): 3 rotations and 3 translations between the pelvis and ground, 3 rotations at the hip (flexion, adduction, and rotation), 3 rotations (flexion, adduction, and rotation) and 3 translations (superior-inferior, anterior-posterior, and medial-lateral) at the knee, and 2 rotations (flexion and eversion) at the ankle. Five degrees of freedom (all 3 translations and adduction and internal rotation) of the patella relative to the femur were locked and patellar flexion was constrained to equal knee flexion.

A pose estimation optimization was used to calculate knee kinematics for each walking trial consistent with the knee contact force measurements [15]. Each cost function evaluation involved adjusting the pose parameters (femoral component position and orientation relative to tibial insert) in an elastic foundation (EF) contact model of the subject's tibiofemoral joint. First, an inverse kinematic analysis was performed in OpenSim where all knee DOFs were locked except for the flexion angle. Next, starting from this motion, a pose estimation optimization was used to determine the superior-inferior translation, medial-lateral translation, and adduction rotation in the EF contact model required to match the medial and lateral compressive contact forces measured experimentally and a medial-lateral shear contact force of zero. For each pose estimation optimization, the knee flexion angle was locked to the value predicted by the OpenSim inverse kinematics analysis, while the anterior-posterior translation and internal-external rotation were locked to values measured using fluoroscopy. The kinematics determined from OpenSim and the pose estimation optimization were used in an OpenSim muscle analysis to calculate muscle-tendon lengths, muscle-tendon velocities, and muscle moment arms. Inverse dynamic loads were also calculated in OpenSim using these kinematics plus the experimentally measured ground reactions.

2.4 Optimization problem formulation

Static optimization was used to predict leg muscle and knee contact forces for each of the three model calibration approaches evaluated. Approach A used unadjusted literature values for muscle-tendon model parameter values (optimal muscle fiber lengths and tendon slack lengths). Approach B calibrated muscle-tendon model parameter values such that the maximum value of
normalized muscle fiber length over one selected gait cycle was one for each muscle. In these two approaches, moment arms were calculated using the subject-specific OpenSim model (Section 2.3) and experimental muscle activations were not tracked. Approach C used a novel two-level optimization formulation. In the outer level optimization, model parameter values (muscle-tendon plus muscle moment arm and neural control) were adjusted such that the inner level optimization reproduced the experimental knee contact force measurements without knowledge of them. For Approach C, three normal walking trials were used for model calibration purposes and the three remaining trials for testing the calibrated model. The static optimization used to predict leg muscle and knee contact forces was similar for all three approaches. Muscle-tendon units were modeled using a Hill-type musculotendon model possessing a rigid tendon and force-length-velocity properties, where the peak isometric strength of each muscle was set to twice literature values [8]. For each time frame of each gait trial analyzed, six inverse dynamics loads were matched as linear equality constraints: three hip moments (flexion, adduction, and rotation), the knee flexion moment, and two ankle moments (flexion and eversion). These loads were considered to be unaffected by knee contact forces. To ensure that the six inverse dynamic loads could be matched exactly, we included a reserve actuator at each joint with a strength of 0.5 Nm. The cost function minimized the sum of squares of muscle and reserve activations using a quadratic programming algorithm.

The static optimization for Approach C used a slightly different formulation and was the inner level of a two-level optimization method. The Approach C static optimization included additional linear inequality constraints that forced the predicted activations to remain “close” to a linear combination of experimental neural commands. It did not, however, have knowledge of the experimental knee contact force measurements. The outer-level optimization of Approach C adjusted model parameter values such that the inner-level static optimization would predict the correct knee contact forces without knowing them. Design variables for the outer level were the following: scale factors for optimal muscle fiber lengths and tendon slack lengths, moment arm offsets, scale factors for activations of sixteen muscles with associated experimental EMG data, and synergy vector weights for twenty-eight muscles without associated experimental EMG. The cost function for the outer level minimized four sets of terms:

- **Tracking terms**: Model activations tracked muscle activations reconstructed from experimental neural commands (Section 2.2), while model medial and lateral knee contact forces tracked corresponding experimental forces.
- **Bound terms**: Model activations and parameter values were constrained within the following bounds: activations reconstructed from experimental neural commands between 0 and 0.7, moment arm offsets between -5 and 5 mm, and scale factors for optimal muscle fiber lengths and tendon slack lengths between ± 20% of literature values [14].
- **Constraints**: Scale factors for optimal muscle fiber lengths and tendon slack length were constrained to have a maximum deviation of 20%, moment arm offsets and normalized fiber lengths for muscles sharing the same insertion point and exerting a similar function were constrained within 5 mm.
- **Minimization terms**: Muscle passive forces and reserve activations (from inner-level reserve actuators) were minimized.

Calibration for Approach C involved running the two-level optimization using three walking trials simultaneously in the inner level. At each optimization step, all model parameter values (time invariant) were transferred to the inner level to calculate muscle activations for the three calibration trials (Figure 1). Once all model parameter values were calibrated, testing for Approach C involved running only the inner-level static optimization using the calibrated model parameter values with the three walking trials held back for calibrated model testing purposes.
3 RESULTS

3.1 Optimization performance

Musculoskeletal model parameter values (optimal muscle fiber lengths, tendon slack lengths, and moment arms) obtained directly from the literature could not produce realistic muscle activations over all time frames for any of the trials. Therefore, no feasible solutions were reached using Approach A due to excessively high reserve activations ($a_{Res} = 1304 \pm 2873$). Conversely, pre-optimized muscle-tendon model parameter values (Approach B) allowed the static optimization to find reasonable results for all trials using very low values of reserve activations ($a_{Res} = 0.000 \pm 0.002$). Feasible solutions were also found for all trials using Approach C ($a_{Res} = 0.04 \pm 0.11$ for calibration trials and $a_{Res} = 0.11 \pm 1.2$ for prediction trials). For this reason, static optimization outputs were compared only for Approaches B and C in Sections 3.3 and 3.4. The two-level optimization in Approach C required approximately 2 days of CPU time using two 6-core processors Intel Xeon 2.39 GHz processors and 24 GB of RAM. However, the inner-level optimization (i.e., predicting muscle activations in any approach) required just over one second (< 1.2 s) to analyze a complete gait cycle.

3.2 Knee contact forces

Without using muscle synergies or calibrated muscle-tendon model parameter values (Approach A), the predicted knee contact forces were unrealistic. Mean medial contact force was 47.4 times larger than the mean experimental value while mean lateral contact force was 36.2 times larger than the corresponding experimental mean. When pre-optimized muscle-tendon model parameter values were used (Approach B), static optimization led to reasonable total contact force magnitudes. For this approach, medial contact force was predicted with reasonable accuracy for all six walking trials ($R^2 > 0.79$, RMSE < 115 N) (Table 1). In contrast, lateral contact force was predicted with poor accuracy, at times (between 25 and 50% of the gait cycle) producing infeasible results where tensile forces would need to be present in the lateral
compartment. When using knee contact force data to calibrate the model (Approach C), highly accurate knee contact force predictions were obtained for all six gait trials (Figure 2 and Table 1). For this approach, accuracy was high not only for the calibration trials (medial RMSE < 121.7 N, lateral RMSE < 112.8 N, total RMSE < 96.7 N) but also for the prediction trials (medial RMSE < 130.1 N, lateral RMSE < 144.3 N, total RMSE < 161.0 N).

![Figure 2. Mean knee contact force predictions. Black solid line represents the mean contact force values of the three gait trials and the grey surface two standard deviations. Dotted blue curves represent the mean values obtained in Approach B and the dashed red curves the mean values obtained in Approach C. Calibration trials were the three gait trials in which the model was calibrated in Approach C and prediction trials were the other three gait trials.](image)

<table>
<thead>
<tr>
<th></th>
<th>Calibration</th>
<th>Prediction</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Medial</td>
<td>Lateral</td>
</tr>
<tr>
<td>Approach B</td>
<td>0.91 ± 0.05 (99.5 ± 16.0)</td>
<td>-2.30 ± 1.48 (290.2 ± 67.6)</td>
</tr>
<tr>
<td></td>
<td>0.89 ± 0.08 (107.1 ± 43.0)</td>
<td>-1.77 ± 0.43 (296.5 ± 34.7)</td>
</tr>
<tr>
<td>Approach C</td>
<td>0.97 ± 0.02 (57.0 ± 19.5)</td>
<td>0.84 ± 0.04 (64.2 ± 7.6)</td>
</tr>
<tr>
<td></td>
<td>0.91 ± 0.03 (96.4 ± 16.7)</td>
<td>0.76 ± 0.12 (85.4 ± 10.3)</td>
</tr>
</tbody>
</table>

3.3 Muscle contributions

Variations in muscle forces between Approaches B and C explained the differences in knee contact force predictions. The main difference between these two approaches was that in Approach C, knee contact forces were tracked in the outer level for the calibration trials. Therefore, the differences in total knee varus valgus (VV) muscle moment contributions and superior-inferior (SI) muscle force contributions were different between the two approaches (Figure 3). Individual muscle contributions to these loads were evaluated for the three calibration trials. Differences in VV muscle moment contributions were higher than 1 Nm for four muscles (gaslat, sart, tfl and vaslat). Differences in SI muscle force contributions were
higher than 25 N for five muscles (gaslat, gasmed, sart, tfl and vaslat) (Figure 4). Differences in VV moment during early stance (first 20% of the gait cycle) were primarily due to the fact that vaslat had a much higher VV contribution in Approach B. During the rest of the stance phase, the lower gaslat and tfl VV contributions explained the differences in the total VV moment. The VV peak moment at 95% of the gait cycle in Approach B was due to changes in gaslat and semiten (although the mean semiten moment contribution difference was lower than 1 Nm). During the first 20% of the gait cycle, SI force was higher in Approach B, mainly due to the higher vaslat contribution in Approach B compared to C. For the rest of the cycle, SI muscle force contributions were higher for Approach C, mainly due to higher contributions from gaslat, sart, and tfl. These observed differences between approaches likely relate to difference in calibrated model parameter values.

Figure 3. Total knee varus-valgus moment and superior-inferior force contributions between approaches for the three calibration trials.

Figure 4. Varus-valgus moment and superior-inferior force contributions for muscles with the greatest differences between approaches for the three calibration trials.

3.4 Model parameter variations

Optimal muscle fiber lengths and tendon slack lengths were higher overall in Approaches B and C compared to Approach A and had high variability (Table 2). For optimal muscle fiber lengths, Approach B values were statistically higher for central muscles and Approach C values higher for medial muscles than in Approach A. For tendon slack lengths, Approach B and C values were statistically higher for all muscles than in Approach A. These differences explain why
Approach A could not find a feasible solution for all time frames. Between Approaches B and C, no statistical differences were observed. The optimal muscle fiber lengths and tendon slack lengths obtained from the literature (Approach A) led to normalized muscle fiber lengths higher than 1.5 for eleven muscles (fdl, fhl, gaslat, gasmed, gem, perbrev, perlong, pertert, piri, soleus, tibpost), representing very high passive muscle force values. For this approach, six of the mentioned muscles had mean passive forces higher than 1000 N, and in three (gasmed, soleus and tibpost) passive forces were higher than 10,000 N, which is unrealistic. For Approaches B and C, all passive forces remained below 200 N. Approach B only had one muscle (soleus) with a mean passive force higher than 20 N, while in Approach C, a mean passive force above 20 N occurred for nine muscles. The higher gaslat passive force would explain the differences in its VV moment and SI force contribution between Approaches B and C observed in Section 3.3.

Given that activation scale factors for Approach C were bounded to be between 0 and 1, these scale factors had high variability ($sa = 0.41 \pm 0.24$ for medial muscles, $sa = 0.53 \pm 0.64$ for central muscles, and $sa = 0.38 \pm 0.31$ for lateral muscles). In Approaches A and B, muscle activations were not tracked, and therefore no activation scale factors were used. The differences in sart and tfl VV moment contributions and SI force contributions between Approaches B and C (Section 3.3) can be explained by changes in muscle activations (Figure A.2, Appendix).

Changes in muscle contributions to inverse dynamics loads also had high variability among muscles. Standard deviation was higher than 1 cm for medial and lateral muscles in the knee flexion moment, for medial muscles in the subtalar moment, and for lateral muscles in the ankle moment. However, only knee superior-inferior offsets for central muscles were statistically different from zero. The differences in vaslat VV moment and gasmed knee SI force contributions (Figure 4) would be explained mainly by their moment arm offsets.

Table 2. Similarity of model parameter values obtained for Approaches B and C relative to Approach A for medial, central, and lateral muscles. Similarities are reported as percent differences for optimal muscle fiber lengths $L^o_M$ and tendon slack lengths $L^o_s$. Statistically significant differences ($p < 0.05$) in mean values between Approaches B and C relative to Approach A are indicated by a star (*).

<table>
<thead>
<tr>
<th>Approach</th>
<th>Medial</th>
<th>Central</th>
<th>Lateral</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L^o_M$ (%)</td>
<td>B</td>
<td>5.0 ± 12.0</td>
<td>10.8 ± 4.9*</td>
</tr>
<tr>
<td>C</td>
<td>6.0 ± 13.9*</td>
<td>8.1 ± 13.7</td>
<td>6.0 ± 14.4</td>
</tr>
<tr>
<td>$L^o_s$ (%)</td>
<td>B</td>
<td>4.9 ± 12.0*</td>
<td>10.7 ± 4.9*</td>
</tr>
<tr>
<td>C</td>
<td>5.7 ± 14.4*</td>
<td>8.5 ± 6.1*</td>
<td>10.0 ± 12.5*</td>
</tr>
</tbody>
</table>

Table 3. Moment arm offsets obtained in Approach C. Values statistically different from zero are indicated by a star (*). All offsets are reported in mm except for the knee superior-inferior force moment arm, which is dimensionless.

<table>
<thead>
<tr>
<th></th>
<th>Medial</th>
<th>Central</th>
<th>Lateral</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hip flexion</td>
<td>-0.0 ± 6.0</td>
<td>12.3†</td>
<td>-0.0 ± 3.6</td>
</tr>
<tr>
<td>Hip adduction</td>
<td>-1.3 ± 6.5</td>
<td>10.14†</td>
<td>2.1 ± 3.5</td>
</tr>
<tr>
<td>Hip rotation</td>
<td>0.4 ± 3.3</td>
<td>-1.9†</td>
<td>0.3 ± 7.0</td>
</tr>
<tr>
<td>Knee flexion</td>
<td>3.7 ± 11.7</td>
<td>12.7 ± 5.7</td>
<td>3.6 ± 12.4</td>
</tr>
<tr>
<td>Knee adduction</td>
<td>-1.3 ± 6.9</td>
<td>4.0 ± 5.3</td>
<td>-2.6 ± 7.3</td>
</tr>
<tr>
<td>Knee sup-inf</td>
<td>0.01 ± 0.03</td>
<td>-0.03 ± 0.00*</td>
<td>-0.00 ± 0.01</td>
</tr>
<tr>
<td>Subtalar</td>
<td>-1.3 ± 11.1</td>
<td>5.4 ± 7.9</td>
<td>-5.0 ± 6.6</td>
</tr>
<tr>
<td>Ankle</td>
<td>-8.4 ± 5.3</td>
<td>-2.4 ± 4.6</td>
<td>-11.1 ± 11.1</td>
</tr>
</tbody>
</table>

†Only one value
4 DISCUSSION

The goals of this study were two-fold. First, we wanted to investigate how model calibration differs when knee contact force data are not used in calibration process (the most common case, Approaches A and B) and when they are used (Approach C). Second, we wanted to evaluate if a set of model parameter values that led to accurate contact force predictions for some walking trials (calibration trials) could predict knee contact forces with comparable accuracy for other walking trials (prediction trials). Approach A used muscle-tendon model parameter values taken directly from the literature [14], whereas in Approach B these parameter values were pre-calibrated. In Approach C, apart from calibrating muscle-tendon model parameter values, we modified skeletal (moment arms) and activation (muscle synergy components) parameter values using a two-level optimization. Using the latter approach, a set of model parameter values was obtained that led to highly accurate knee contact force predictions for the three testing trials. Differences in the predicted knee contact forces and leg muscle forces between the three approaches suggest that poor calibration of neuromusculoskeletal model parameter values may be a primary contributing factor to inaccurate prediction of these internal forces.

While muscle-tendon model parameter values obtained from the literature provide an estimate of the magnitude of these parameter values, they can lead to infeasible static optimization results. For example, Approach A predicted excessively high passive muscle forces. An important finding was that pre-calibrating muscle-tendon model parameter values to make normalized muscle fiber lengths operate on the ascending region of the normalized force-length curve (Approach B) [7], and maintaining these parameter values close to the literature ones, improved knee contact force predictions substantially. In fact, medial contact force predictions for Approach B were surprisingly accurate in terms of both shape and magnitude. However, lateral contact force predictions were still poor, as has been the case in previous studies [16]. The main differences in knee contact predictions between Approach B and Approach C, where neuromusculoskeletal parameter values were calibrated (using knee contact force information in Approach C), can be summarized by changes in five muscles: three lateral muscles (gaslat, tfl, and vaslat) and two medial muscles (gasmed and sart).

Muscle forces obtained in Approaches B and C were similar in magnitude and shape to those predicted in other studies [2,17,18]. Only minor differences were observed for some muscles, such as a lower gmed force in our study compared to in [2]. Nonetheless, overall, all predicted muscle force magnitudes were within the ranges reported in the literature [19].

The main limitation of this study was that all optimizations were carried out using the same movement task, which was overground walking at self-select speed. Using other types of movements, for instance trials were the five muscles mentioned above played a more important role, may lead to a better calibration when no knee contact force data are available (the most usual case) and consequently to better contact force predictions. In addition, only one subject was tested, and trying the three calibration approaches with other subjects would generalize our conclusions. Future research will also explore new ways to introduce more constraints into the static optimization problem formulation.

To conclude, our main recommendation for calculating muscle forces using static optimization is to ensure that muscles operate on the ascending region of their normalized force-length curves. However, such an approach does not ensure that the predicted muscle forces will be correct. We also observed that it was possible to obtain a single set of neuromusculoskeletal model parameter values that predicts accurate knee contact forces for walking trials not used in the calibration process. Further research should be carried out to develop better model calibration methods when no knee contact force data are available.
5 REFERENCES


APPENDIX

Figure A.1. Muscle forces for all muscles in Approaches B and C.

Figure A.2. Muscle activations for all muscles in Approaches B and C.
Towards bridging the gap between motion capturing and biomechanical optimal control simulations

Tobias Gail\textsuperscript{*}, Ramona Hoffmann\textsuperscript{†}, Markus Miezal\textsuperscript{#}, Gabriele Bleser\textsuperscript{#}, Sigrid Leyendecker\textsuperscript{*}

\textsuperscript{*} Chair of Applied Dynamics
University of Erlangen-Nuremberg
Erlangen, Germany
\textsuperscript{†} née Maas, Mathematical Methods in Dynamics and Durability
Fraunhofer Institute for Industrial Mathematics (ITWM)
Kaiserslautern, Germany
\textsuperscript{#} Department Computer Science
Technical University Kaiserslautern
Kaiserslautern, Germany
\textsuperscript{#} [miezal, bleser]@cs.uni-kl.de
\textsuperscript{*} sigrid.leyendecker@ltd.uni-erlangen.de
\textsuperscript{†} ramona.hoffmann@itwm.fraunhofer.de

ABSTRACT
Within this work, we make a first attempt towards improving human motion capture by combining motion capturing measurements and optimal control simulations of a human steering motion. We start with measurements obtained from a stationary optical system, a widespread capturing technology in biomechanics and movement science, under laboratory conditions. From an optimal control point of view, the goal is to increase the realism of simulated human motion through measurements. From a motion capturing point of view, the goal is to compensate for measurement sparsity, errors or lacks through meaningful assumptions based on biomechanical simulation. Our preliminary results show that a fusion of physical laws, biomechanical simulation and real data within an optimal control simulation framework indeed have the potential to improve motion capture and synthesis with respect to some of their inherent problems.

Keywords: optimal control simulation, optical motion capture, soft tissue artefacts, measurement sparsity.

1 INTRODUCTION
The abilities to capture and to simulate human motion are important enabling technologies for a wide range of applications in the professional as well as the private sector, ranging from health and ergonomics over human-machine-interaction to sports and games \cite{15, 20, 4}. Different interdisciplinary research branches of computer science, mechanical engineering and mathematics are concerned with one or the other direction, i.e. with accurate and robust capturing or with realistic and physically plausible synthesis of human motion. Some of the most related disciplines include computer graphics, robotics and biomechanics. Though human motion capture and synthesis have been extensively studied and used, they are still subjects of research in the different domains. Moreover, this research often happens without bringing together the foundational knowledge, approaches and ideas that the different disciplines can offer.

There are methodically very different approaches to include optical measurement data in the simulation of human motion. Many are based on forward dynamics simulations, where, e.g., the motion is constrained to follow the data followed by an inverse dynamics analysis to determine the corresponding control, see, e.g., \cite{5}. In \cite{16}, forward dynamic simulation is combined with an outer optimisation loop to find parameters producing a simulation that is most consistent with the data. Furthermore, a parametric control model is fitted to the data in \cite{19}, using the gains to identify the specific linear quadratic regulator cost function for a bicycle rider.

Within this work, we make a first attempt towards bridging this gap by investigating the specific scenario of combining motion capturing measurements of a human steering motion and optimal control simulations of this motion. While we aim at moving towards ambulatory motion capturing
systems in the long run, we start our investigation with a stationary optical systems, which is a widespread capturing technology in biomechanics and movement science. From a motion capturing point of view, the goal is to compensate for measurement errors or lacks and to reduce the required amount of measurements, while at the same time maintaining high precision, robustness and realism, all through meaningful assumptions based on biomechanical simulation. From an optimal control point of view, the goal is to resolve ambiguities and to personalise and increase the realism of simulated human motion guided through measurements.

1.1 Optical motion capture

Marker-based optical motion capture systems are typically based on a stationary infrastructure of high-resolution and high-speed infrared cameras, which are calibrated and synchronised with respect to each other. Retroreflective markers are then attached to specific human anatomical landmarks. These markers are tracked throughout the images of the multiple cameras and their trajectories are estimated in 3D space. Assuming some type of body model, most often a kinematic chain model [23], the motion of the subject is derived from the set of marker trajectories. Obviously, this requires (1) matching the dimensions of the digital model with the anthropometry of the human subject and (2) matching the digital marker positions with the real placement on the subject. The alignment is usually based on (functional) calibration, measurements and anthropometric tables, e.g., [17]. Available optical motion capture systems, e.g., from Vicon or NaturalPoint, are rather expensive, but widespread in biomechanics and ergonomics applications.

While the precision of measuring the marker positions can be extremely high, depending on the resolution and positioning of the cameras, there are a number of inherent problems and proposed solution paths: First, the sparsity of measurements in temporal and spatial domain and the risk of measurement lacks due to occlusions can be reduced by optimizing the camera placement and increasing the number of cameras, the camera update rate and the number of markers. However, this trades off with costs, setup time and wearing comfort. Second, deriving the underlying movement from the marker trajectories is usually based on the assumption that there is no relative motion between the external surface, where the markers are placed, and the modelled bone structure. Soft tissue artefacts, and further, simplifications of the body model and calibration errors, however, lead to violation of this assumption and can cause significant errors. A whole body of research tries to tackle this problem, e.g., based on new calibration procedures in combination with redundant marker sets (cluster markers) or more detailed biomechanical body models [9, 18, 6]. However, this often results in impractical setup procedures, e.g., a high number of markers to be positioned on the body or specific calibration infrastructure.

Instead of heavily relying on the measurements and trying to improve their quality, another promising approach is to embed more intelligence into the digital models, e.g., by including physics-based simulation, learnt motion models or higher-level knowledge about human motion [22, 7, 1]. The above approaches have already been used to improve optical motion capture, also based on monocular and multi-view markerless settings and when moving towards low-end and consumer systems.

1.2 Optimal control simulation

In general, an optimal control problem aims to find a trajectory of state and control variables, minimising an objective functional and fulfilling constraints in the form of differential equations as well as algebraic equalities and inequalities. The constraints represent the system’s behaviour (e.g., its dynamics) as well as boundary conditions and path constraints, [8]. A solution to this mathematical model is approximated by a numerical simulation method [21, 3]. One problem of this abstract description is that the solution is found merely based on the technical model. When the problem is transformed (e.g., as in our approach) to a constrained optimisation problem, the optimising algorithm has no intuition on what is a natural and biologically realistic human motion. Human motion is controlled by the central nervous system taking into account an immensely
complex variety of aspects which are difficult to include in a mathematical description amenable to a simulation in acceptable time.

Another problem is the local nature of optimal control solutions (global optimisation is usually too demanding in terms of computational costs), thus the determined motion depends on an initial guess, which might be hard to find in the infinite number of possibilities to move from an initial to a final state. This is an obvious point where the combination with measurement data can be used to improve the efficiency and realism of human motion simulation. However, apart from the initial guess, the inclusion of data is possible in various parts of the optimal control problem formulation as is discussed in this work.

1.3 Our approach

We propose combining motion capture technology and optimal control simulations in order to tackle their inherent problems, i.e., compensate for measurement sparsity, lacks and errors, as well as shortcomings of digital body models and the curse-of-dimensionality related to the simulation space, while providing a realistic, natural and high-precision motion trajectory. We start with integrating measurements from a marker-based optical system into optimal control simulations based on a constrained optimisation framework [12]. In this first step, we investigate in particular the following aspects considering a human steering motion: (I) How should the measured marker positions be incorporated into the optimisation, i.e., in the objective function as so-called soft constraints or as hard constraints to the optimisation? (II) When combining the measured marker positions with a physiologically motivated cost function, how far can the measurement update rate be reduced while still obtaining accurate, realistic and natural results?

In order to clarify these questions, we follow a systematic approach, performing a sequence of numerical experiments: First, we investigate (I) by considering only measured marker positions and the equations of motion. The result is then called feasible solution. Second, we combine the measured marker positions and the equations of motion with different established physiologically motivated terms in the cost function, in particular minimal torque and minimal torque change. These are then called minimal torque solution and minimal torque change solution. Third, we investigate (II) by gradually reducing the measurement update rate and looking at the resulting deviation of the simulated motion from the reference motion. For each solution, the reference motion is defined as the simulated motion, where all optical measurements are used (i.e., 100 Hz). The described deviation is used to assess the accuracy of the resulting motion. Moreover, realism and naturalness are assessed via the residual error with respect to the marker positions and the evolution of the joint torques.

The preliminary results show that a fusion of physical laws (i.e., the equations of motion), biomechanical simulation (i.e., physiologically motivated cost functions) and real data (i.e., optical marker positions) within an optimal control simulation framework indeed have the potential to improve motion capture and synthesis with respect to the above-mentioned inherent problems.

2 MEASUREMENTS

Optical measurements are captured during different trials of a steering motion from one 30 year old male subject (height 176 cm) using the commercially available NaturalPoint OptiTrack system with twelve infrared cameras positioned on two heights around the steering wheel mockup. Two rigid bodies (with three retroreflective markers each) and six individual markers were attached to the subject’s right shoulder, arm and hand: one rigid body on the shoulder and one on the back of the hand (above an inertial measurement unit, IMU), one marker on the upper arm, three around the elbow (close to lateral/medial epicondyl and elbow tip) and two on both sides of the wrist (close to ulnar and radial styloid processes). The markers were attached to a tight velcro jacket delivered with the system. After calibrating the optical system and measuring the dimensions of the subject and the positions of the markers with respect to the segments, steering motions were
Table 1. Means and standard deviations in meters of pairwise marker distances with respect to markers on rigid segments. In an ideal world, the distances should be constant.

<table>
<thead>
<tr>
<th></th>
<th>mean</th>
<th>standard deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lateral/medial epicondyl</td>
<td>0.09</td>
<td>3.4e-05</td>
</tr>
<tr>
<td>elbow tip/ulnar styloid</td>
<td>0.26</td>
<td>5.12e-04</td>
</tr>
<tr>
<td>elbow tip/radial styloid</td>
<td>0.28</td>
<td>5.73e-04</td>
</tr>
<tr>
<td>ulnar/radial styloid</td>
<td>0.10</td>
<td>5.58e-04</td>
</tr>
</tbody>
</table>

performed at different speeds while keeping the shoulder stationary. The 3D positions for the individual markers and the 3D positions and orientations for the two rigid bodies were captured at 100 Hz. Note that the hand’s rigid body position was corrected for the offset due to the IMU.

In the present study, only linear position information is used and the markers on the upper arm are also excluded, since the shoulder is assumed fixed. Hence, six marker positions are modelled (cf. Section 3). Moreover, for the experiments described in Section 4, a data sequence containing 50 sample points (0 to 0.49 seconds) showing a quarter turn with the right hand on the steering wheel mockup was extracted. Figure 1 shows the measurement setup. For future experiments, we also captured inertial measurements at 100 Hz from two IMUs attached to the upper and forearm. However, the data is not used in the present study. Note, when analysing the 3D marker positions obtained from the optical system (e.g., by looking at the variations of pairwise marker distances on rigid segments), errors due to soft tissue artefacts can be noted (cf. Table 1), which motivates the investigation of this paper. Further observations regarding soft tissue artefacts are included in Section 4.

3 OPTIMAL CONTROL SIMULATION

3.1 Human arm model

For the simulation, the human arm is modelled as a multibody system consisting of three rigid bodies. A cylindrical upper arm is fixed in space by a spherical joint representing the shoulder. The elbow and wrist are modelled as cardan joints connecting the cylindrical forearm to the upper arm and the parallelepiped shaped hand to the forearm, respectively (cf. Figure 2). The bodies’ dimensions are personalised for the subject and the optical marker positions are placed manually in the model based on measurements as explained in Section 4.1. Thus, the exact definition of the personalised model is already a result from the measured data.
3.2 Optimal control problem and simulation

Two inherently different approaches for the solution of an optimal control problem are the so-called indirect (first optimise then discretise) and direct (first discretise then optimise) approach, see e.g., [21, 3]. In this work, a direct transcription method called discrete mechanics and optimal control for constrained systems (DMOCC), see [11, 14], falling into the latter class, is used to approximate the solution at the time nodes \( t_0, t_1 = t_0 + h, \ldots, t_N = t_0 + Nh \) on an equidistant time grid with time step \( h \).

As described in detail in [2], the kinematic description of the rigid multibody system is based on a redundant configuration variable \( q_k \in \mathbb{R}^{36}, k = 0, \ldots, N \) consisting of the placement of the centre of mass and the orientation represented by three directors that are aligned with the principal axes of inertia for each rigid body, respectively. A set of 29 holonomic constraints ensure orthonormality of each body’s directors (thus they represent the columns of a rotation matrix) as well as the coupling by the joints, thus the complete model has 7 degrees of freedom.

A nodal reparametrisation \( F_d : \mathbb{R}^7 \rightarrow \mathbb{R}^{36} \) updates the redundant configuration \( q_{k+1} = F_d(u_{k+1}, q_k) \) for \( k = 0, \ldots, N - 1 \) in terms of discrete generalised coordinates \( u_d = \{u_k\}_{k=1}^N \) with \( u_k \in \mathbb{R}^7 \) such that the constraints are fulfilled. In contrast to a formulation in terms of minimal coordinates (joint angles) from the beginning, this procedure ensures that rotations are always small and thus avoids the danger of singularities. The configuration variable \( q \) can be treated in a linear space, yielding a Lagrangian function with a constant mass matrix. A structure preserving scheme (symplectic-momentum with good energy behaviour) approximates the dynamics. It is derived via a discrete variational principle, see [13], where a discrete Lagrangian \( L_d : \mathbb{R}^{29} \times \mathbb{R}^{29} \rightarrow \mathbb{R} \) approximates the action in one time interval. The discrete Euler-Lagrange equations resulting from the stationary condition for the discrete action are reduced to minimal dimension using a discrete null space matrix \( P(q_k) \in \mathbb{R}^{29 \times 7} \) and the nodal reparametrisation \( F_d \) resulting in (2), see [2, 10]. They involve the left and right control forces \( f^{k-1}_l = B(q_k) \cdot \tau_{k-1}, f^k_r = B(q_k) \cdot \tau_k \in \mathbb{R}^{36} \) which are computed from the discrete generalised controls \( \tau_d = \{\tau_k\}_{k=0}^{N-1} \) with joint torques \( \tau_k \in \mathbb{R}^7 \) (assumed to be constant during one time interval) using the input transformation matrix \( B(q_k) \in \mathbb{R}^{7 \times 29} \), see [11] for a detailed introduction to DMOCC.

The optimal control problem is simulated solving the following nonlinear constrained optimisation problem using an SQP algorithm in Matlab. Minimisation of the objective function \( J_d \)

\[
\min_{u_d, \tau_d} J_d(u_d, \tau_d)
\]  
subject to the fulfilment of the discrete equations of motion

\[
P^T(q_k) \cdot \left[ D_2 L_d(q_{k-1}, q_k) + D_1 L_d(q_k, f(u_{k+1}, q_k)) + f^{k-1}_l + f^k_r \right] = 0
\]  
boundary conditions

\[
s(u_d, \tau_d) = 0
\]
and path constraints

$$h(u_d, \tau_d) \leq 0$$

(4)

We perform optimal control simulations with different objective functions and path constraints. In order to address question (I), the measured marker positions are part of the objective function of the first problem in Section 3.2.1. This is sometimes called the inclusion of data as soft constraints and differs essentially from their appearance as hard constraints in Sections 3.2.2, 3.2.3 and 3.2.4, where the latter two involve commonly used physiologically motivated cost functions.

3.2.1 Minimisation of marker position residual errors

The goal of the first optimal control problem is to minimise the residual errors between the measured marker positions $$\bar{m}_k$$ and the simulated marker positions $$m_k$$ at all data points, thus the objective function

$$J_d(u_d, \tau_d) = \sum_{k=0}^{N} (m_k - \bar{m}_k)^T \cdot (m_k - \bar{m}_k)$$

is minimised subject to the discrete equations of motion (2) and boundary conditions (3), while no further path constraints are present.

3.2.2 Feasible trajectory

State and control trajectories that are feasible in the sense that they fulfil the equations of motion (2) and boundary conditions (3) are obtained by minimising the objective

$$J_d(u_d, \tau_d) = 1$$

with the additional path constraints (4) imposing an upper bound \( \varepsilon \in \mathbb{R} \) on the marker position’s residual errors taking the form

$$h(u_d, \tau_d) = \sum_{k=0}^{N} (m_k - \bar{m}_k)^T \cdot (m_k - \bar{m}_k) - \varepsilon$$

(5)

3.2.3 Minimisation of torque

The third problem minimises the control effort

$$J_d(u_d, \tau_d) = \frac{\Delta t}{2} \sum_{k=0}^{N-1} \tau_k^T \cdot \tau_k$$

subject to the discrete equations of motion (2), boundary conditions (3) and path constraints (5).

3.2.4 Minimisation of torque change

In the last problem, the temporal torque change is minimised, thus

$$J_d(u_d, \tau_d) = \frac{\Delta t}{2} \sum_{k=0}^{N-2} \left\| \frac{\tau_{k+1} - \tau_k}{\Delta t} \right\|^2$$

while the discrete equations of motion (2), boundary conditions (3) and path constraints (5) are fulfilled.

4 RESULTS

Simulation experiments are carried out based on the extracted data sequence as described in Section 2 with the different optimal control problems introduced in Section 3. Here, we first make some remarks concerning optimal control simulation (and its initialisation) with regard to aspect (I) mentioned in Section 1. Then, observations on the soft tissue artefacts are discussed. Finally, addressing (II), we present results from simulations based on gradually reducing the measurement update rate of the recorded data.
4.1 Simulation of the optimal control problem

As mentioned in Section 3.1, the personalised model results from the measurements, more precisely from the measurement \( \bar{m}_0 \) at the first pose at \( t_0 = 0 \). Geometrical data like lengths and radii of the rigid bodies, placement of joints and joint axes as well as the marker placement on the segments and their initial configuration (placement and orientation) are obtained from \( \bar{m}_0 \) via a geometric and inverse kinematic analysis. All numerical experiments start in this pose at rest, which is the only boundary condition (3). Note that no end point condition is formulated since the motion did neither end at rest nor at a known velocity and for a measurement rate of 100 Hz in combination with the chosen time grid with a time step of \( h = 0.01 \text{s} \) and \( N = 49 \), thus 50 measurements \( \bar{m}_0, \ldots, \bar{m}_N \), the number of measurements and simulated configurations coincides such that the residual error at the final point \( \bar{m}_N - \bar{m}_N \) controls the final configuration. Based on the obtained geometrical data, for all experiments, the initial guess for the optimisation parameters \( u_d \) is obtained from the measured marker positions \( \bar{m}_k, k = 1, \ldots, N \) via inverse kinematics. The initial guess for the joint torque trajectory \( \tau_d \) is zero everywhere.

First, the state and control trajectories minimising the marker position residual errors described in Section 3.2.1 are approximated. The resulting configuration trajectory fits the measured data up to a residual error of \( \varepsilon = 0.0218 \), which is used as the upper bound in the inequality path constraints (5) of the following experiments. The largest residual errors occur around the elbow towards the end of the motion. While in the later experiments, the torque evolutions are qualitatively and quantitatively similar in large parts of the motion, see Figures 7, 8 and 9, the torque evolution in this experiment is more jerky and assumes absolute values that are up to 1.75 times larger. Using the number of SQP iterations required by the optimiser until convergence as an indicator for efficiency, this experiment is by far the most inefficient, requiring approximately 20 times more iterations than the feasible solution and approximately 10 times more than the torque minimising and torque change minimising simulations.

For the feasible, torque minimising and torque change minimising simulations described in Sections 3.2.2, 3.2.3 and 3.2.4, the initial guess is determined as described above when all measurements, i.e., a measurement rate of 100 Hz, are used. However, when reducing the measurement update rate, only every \( n \)th measurement with \( n \in \{2, \ldots, 13\} \) is taken into account in Equation (5) and \( \varepsilon \) is reduced accordingly. Consequently, the poses determined by inverse kinematics from the measurements \( \bar{m}_k, k = 1, \ldots, N \) are used either as a piecewise constant initial guess or the configuration \( q \) is interpolated linearly. Note that the last measured data point which is included in all reduced rate experiments is \( \bar{m}_{39} \).

For the three objective functions (feasible, minimising torque and minimising torque change), the two initial guess strategies (piecewise constant and piecewise linear) and the 13 measurement rates (every \( n \)th measurement with \( n \in \{1, \ldots, 13\} \)), a total of 78 optimal control simulations is evaluated in the sequel.

4.2 Soft tissue artefacts

In addition to the soft tissue artefacts discussed in Section 2 that can be observed directly in the measurement data (cf. Table 1), when investigating the residual errors with respect to the marker positions for the three numerical experiments (full measurement rate) in Figure 3, most dominant errors are visible for the three markers around the elbow, with an additional trend of accumulating error from the start to the end of the motion. This can be explained by soft tissue artefacts during the turning motion, where the velcro jacket, to which the markers are attached, slightly slides over the skin. When being solely based on measurements (i.e., ignoring physics-based and biomechanical constraints), such artefacts would result in an erroneous motion, in this case underestimating the amount of turning.
Figure 3. Residual errors with respect to marker positions (distance between simulated and measured marker position for each sample point) at 100 Hz measurement rate.

4.3 Simulations with reduced measurement rate

Reducing the measurement update rate as explained in Section 4.1 results in a range between 100 and 7.69 Hz. Figures 4 and 5 show the error statistics for the tested measurement rates based on a piecewise constant and a piecewise linear initial guess, respectively. Error is here defined as the Euclidean distance between the configurations \( q \) obtained from the reference simulation (using a measurement frequency of 100 Hz in the feasible, torque minimising and torque change minimising simulation) and the simulation with reduced measurement update rate at each time node \( t_k, k = 0, \ldots, N \). The figures then report the average error and its standard deviation for a complete simulation (y-axis) over the measurement rate (x-axis). Each vertical line (illustrating the standard deviation) represents one successful simulation. Missing vertical lines indicate a simulation failure, e.g., due to divergence of the SQP algorithm. Note that errors are computed only in the time interval \([0, 0.39] s\), since \( \bar{m}_{39} \) is the last measurement used in all simulations. When comparing Figures 4 and 5, in particular the close-ups, it can be observed that some simulation experiments have failed below 20 Hz when using a piecewise constant initialisation. Hence, linear interpolation enables convergence at very low measurement update rates. Moreover, it can be seen in the figures that linear interpolation also reduces the overall errors. This is particularly visible for the feasible solution.

When looking at Figure 5, as expected, the errors and standard deviations increase with reduced measurement update rate for all solutions. However, the error of the feasible solution is consistently higher than the error of the solutions including a physiological cost function term. Below 20
Hz, errors and standard deviations of the feasible solution increase significantly and the last two simulations even fail. In contrast, minimising torque and torque change show a better behaviour. Minimal torque shows the lowest error until 12.5 Hz, however, then starts to increase significantly, and fails for the last simulation at 7.69 Hz. Minimal torque change successfully converges for all experiments and shows a comparably small error increase even at the lowest measurement frequencies. These observations confirm that, in our experimental settings, biomechanical simulation can compensate for low measurement update rates. Moreover, while the torque minimising solution provides the most accurate results down to a certain measurement rate, the torque change minimizing solution provides acceptable accuracy for even lower measurement update rates, hence adds further stability. To provide more detailed insights, Figure 6 illustrates the error evolution in $q$ (configuration error) for two concrete measurement update rates (using linear interpolation for the initial guess). We chose 33.3 Hz (every 3rd measurement) as representative for a mid-range frequency and 8.33 Hz (every 12th measurement) as lowest frequency with results for all solutions. These figures confirm the above observations. In addition, it is nicely visible in Figure 6(b), that the errors show a periodic pattern induced by the measurement update rate, i.e., the error becomes lower around measurement points. This is most apparent for the feasible solution, but also clearly visible for the minimal torque solution. Interestingly, the minimal torque change solution, even if showing a slightly higher error than the minimal torque solution for the mid-range frequency, is much less affected by the measurement points. This might indicate a higher independence from measurements, a better ability to deal with errors in these and, as a result, a higher robustness. On the other hand, also error of the time stepping equations (with respect to an analytical solution) grows as the simulation advances in time. This point needs further investigation in the future. One

Figure 5. Mean errors and standard deviations based on piecewise linear interpolation for initial guess.

Figure 6. Error evolution in configuration $q$. 
can also observe that the error tends to decrease throughout the simulation, in particular for the low measurement update rate and the minimal torque and torque change solutions.

Moreover, the evolutions of the joint torques in shoulder, elbow and wrist are shown for the three numerical experiments in Figures 7, 8 and 9, respectively. Note that only the torque evolution in the time interval [0, 0.39]s can be compared, since \( \bar{m}_{39} \) is the last measurement used in all simulations. Two observations can be made here: First, when decreasing the frequency from 100 to 8.33 Hz, the resulting torques of the feasible and minimal torque solution change more significantly than the minimal torque change solution. This is most visible for the wrist. Second, with reduced measurement update rate, the feasible and minimal torque solutions show jerky torque changes, though the actual motion is smooth. Again, this is most apparent for the wrist. In contrast, the minimal torque change solution shows smooth torque changes. This is not surprising, since minimal torque change is an optimisation criterion. However, even when not having ground truth torque at hand, together with the first observation, the smooth torque evolution seems more consistent and plausible, being the origin of a natural motion. Hence, the minimal torque change solution seems to provide torques, which are less dependent on measurements and which indicate a more natural human motion, which are both wanted effects of our study.
5 CONCLUSIONS

Since the inclusion of the measurements as soft constraints by minimising the residual errors in the objective function turns out to be way more inefficient than their inclusion as hard constraints, the purpose of the first numerical experiment is mainly to determine the bound $\varepsilon$ for the later experiments. Also considering that the measurements are a priori known to have errors indicates that it is more promising to use them as guiding points and to define an environment around them where a solution of a biomechanical simulation with a physiologically motivated objective is to be found. From the investigated objective functions here, minimising torque change shows the most realistic and natural results and the highest stability with respect to the reduction of the measurement frequency.

The most obvious future task is to confirm these observations with different types of motions and other physiologically motivated objective functions. Furthermore, it is worthwhile to take into account also measurements of the bodies orientation instead of only marker point positions and information on the accelerations and angular velocities measured from inertial sensors. In particular, the latter means not merely the inclusion of more and different type of data, but the move towards ambulatory motion capturing may overcome many of the shortcomings of optical stationary systems discussed in the introduction and enable a wide range of applications outside the lab. However, also many technical aspects need to be investigated. If there is knowledge on the precession or error-proneness of certain measurements, weighting factors can be introduced accordingly. Secondly, not only the measurement rate, but also the number of marker positions can be reduced, excluding e.g., those with most soft tissue artefacts.

Finally, the inclusion of further information known about the considered motion, like the presence of obstacles in the environment or contact to the surroundings, as e.g., of the hand moving on a circle due to its contact to the steering wheel, may help to increase the realism and naturalness of the simulated motion.

REFERENCES


Underactuated approach for the control-based forward dynamic analysis of acquired gait motions

Francisco Mouzo*, Urbano Lugris*, Rosa Pamies-Vila#, Josep M. Font-Llagunes#, Javier Cuadrado*

* Lab. of Mechanical Engineering
University of La Coruña
Escuela Politecnica Superior
Mendizabal s/n, 15403 Ferrol, Spain
francisco.mouzo@udc.es

# Dept. of Mechanical Engineering
Universitat Politècnica de Catalunya
Diagonal 647, 08028 Barcelona,
Catalonia, Spain
rosa.pamies@upc.edu

ABSTRACT

This paper presents a method to carry out the analysis of acquired gait motions through a control-based forward dynamic approach. Unlike some well-established control-based methods that consider inputs for all the system degrees of freedom, the current work proposes the more realistic alternative of having inputs at joint level only, thus leading to an underactuated system. The ground reactions come from a foot-ground contact model which is built in a pre-processing stage. Different sets of outputs to be tracked by the joint controllers are evaluated. It is observed that choosing as outputs the weighted trajectories of all the system degrees of freedom yields satisfactory results, and that including the weighted ground reactions provides only marginal improvement.

Keywords: Biomechanics, Gait, Control-based forward dynamics, Foot-ground contact model, Underactuated system.

1 INTRODUCTION

The analysis of acquired gait motion through forward dynamics instead of traditional inverse dynamics offers certain advantages, such as superior dynamic consistency [1], ability to consider muscle activation and contraction dynamics when descending at muscular level [2], and feasible computation of contact forces between the subject and the ground or assistive devices [3]. Furthermore, since the forward dynamic analysis implies the integration in time of the model equations of motion, it must face the inherent challenges of gait dynamics (intermittent contact, stability, etc.), and can be perceived as an intermediate step to motion prediction, having less uncertainty as the resulting motion is known.

In a previous work [4], the authors addressed the forward dynamic analysis of an acquired gait motion by means of trajectory tracking controllers associated to all the degrees of freedom of the model. It was shown that the computed torque control (CTC) method provided good accuracy and was extremely robust with respect to the selected gain values. In fact, the computed muscle control (CMC) proposed in [2] at muscular level, uses the CTC method at joint level (introducing then an optimization loop to compute the muscular excitations that generate the obtained joint torques).

However, the human body is not a fully-actuated system, but an underactuated one. Therefore, if an approach closer to reality is sought, it is not admissible to control the six degrees of freedom of the base body (usually, the pelvis or the foot in contact with the ground). Instead, actuators can only be associated to human joints, while the external reactions coming through the feet can be represented by foot-ground contact models. An example of this approach can be found in [5] for a jumping exercise.

In this work, the forward dynamic analysis of an acquired gait motion is performed, considering the human body as an underactuated system and, hence, placing controlled actuators at the joints only. Controllers are supposed to track a number of outputs, which can be trajectories, forces or a combination of both. The objectives are to check whether the acquired motion can be
approximated through this method, and to determine which is the specific implementation that best reproduces the motion, the joint drive torques and the ground reactions. The authors believe that gaining insight into this problem will be helpful for the more challenging topic of motion prediction.

The remaining of the paper is organized as follows. Section 2 presents the experiment data and the human model, including the foot-ground contact model. Section 3 describes the proposed method to carry out the control-based forward dynamic analysis of the acquired motion. Section 4 explains different alternatives of output choices and the corresponding results. Finally, the conclusions of the paper are drawn in Section 5.

2 EXPERIMENT AND MODEL

Gait data from a healthy adult male, 27 years old, mass 84 kg and height 1.75 m has been taken from the Library of Computational Benchmark Problems [6] developed by the IFToMM Technical Committee for Multibody Dynamics. The benchmark problem, named Gait 2D, provides the histories of the markers used to optically capture the motion of the mentioned subject, along with the ground reactions measurements provided by force plates. Furthermore, it gives the parameters defining the 12-segment, 14-degree-of-freedom planar human model shown in Figure 1 (right), used by the benchmark authors to perform an inverse dynamic analysis to derive the joint drive torques that generated the motion and the motion-consistent ground reactions (slightly different from the measured ones). The histories of these magnitudes, as well as the histories of the Cartesian coordinates and angles of the planar model in Figure 1, are also supplied.

For the purpose of conducting the forward dynamic simulation of the mentioned planar model, the state-space matrix-R formulation [7] has been applied,

\[ M\ddot{z} = Q + Bu \]  

where \( M \) is the system mass matrix, \( \ddot{z} \) is the vector of second time-derivatives of the coordinates (accelerations), \( u \) is the vector of actuations (less than the system degrees of freedom), projected to the coordinates space through matrix \( B \), and \( Q \) is the vector of velocity-dependent and remaining applied forces (gravitational and ground reactions).

The configuration vector \( z \) of 14 independent coordinates that has been selected in this work is formed by the two Cartesian coordinates of the hip and the angle between vertical axis and trunk (three degrees of freedom of the base body), along with the 11 relative angles illustrated in Figure 1 (right),
The equations of motion are integrated in time by means of the single-step implicit trapezoidal rule, with the accelerations \( \ddot{z} \) as primary variables.

The foot-ground contact model plays a key role in the proposed problem. If a force model is chosen, the system is certainly underactuated, and control methods for such types of systems must be used, with the additional difficulty of the unstable nature of gait. If a constraint method is selected seeking to have a fully-actuated system at all times, constraints must be alternatively imposed to the feet (thus perturbing the continuous motion they experience during gait, even at the stance phase), and the impact at landing must be dealt with in some way. Consequently, a force model has been used in this work.

Moreover, there is a problem that must be faced when using foot-ground force contact models in the forward dynamic analysis of acquired gait motions: the selection of the contact model parameters and, more importantly, of the feet boundaries. A not sufficiently good location of feet boundaries can yield huge contact forces that make the simulation fail. Therefore, an optimization method to select the mentioned characteristics of the contact model, similar to the one proposed in [8], is required to be applied prior to the forward dynamic analysis as a pre-processing stage, to ensure reasonable contact forces during the simulation.

![Figure 2. Foot boundary definition.](image)

The nonlinear volumetric contact model proposed in [9] has been used, for which the normal and tangential contact forces are defined as,

\[
\begin{align*}
    f_n &= k_h V^h + a_h V v_n \\
    f_t &= -\mu v_c f_n \\
    \mu(v_c) &= \mu_f \arctan(v_c/v_s)
\end{align*}
\]

where \( V \) is the interpenetration volume, \( k_h \) is the hyper-volumetric pseudo-stiffness, \( h \) is an exponent which depends on the volumetric stiffness and geometrical properties, \( a_h \) is the foundation stiffness multiplied by the damping, \( v_n \) and \( v_t \) are the normal and tangential velocities at the centroid of the deformed volume, respectively, \( \mu \) is the friction coefficient, \( \mu_f \) is the asymptotic friction coefficient, and \( v_s \) is a shape factor. The values of these parameters have been taken from [9], except for \( k_h \) and \( a_h \), which have been reduced, since the values in [9] are for spheres and the model in this work is two-dimensional and, hence, it would assume a 1 m length in the direction orthogonal to the plane. The values of all the parameters are listed in Table 1.

\[
z^T = \{ x, y, \alpha_0, \alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5, \alpha_6, \alpha_7, \alpha_8, \alpha_9, \alpha_{10}, \alpha_{11} \}
\]
Table 1. Parameters of the foot-ground contact model.

<table>
<thead>
<tr>
<th>$k_h$</th>
<th>$h$</th>
<th>$a_h$</th>
<th>$\mu_f$</th>
<th>$v_f$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^5$</td>
<td>0.79</td>
<td>$2 \cdot 10^6$</td>
<td>0.34</td>
<td>0.034</td>
</tr>
</tbody>
</table>

The optimization pre-process to set the feet boundaries is as follows. For each foot, a local reference system $(\mathcal{X},\mathcal{Y})$ is defined as shown in Figure 2, with the origin at the ankle and the $x$-axis horizontal in the support position. Then, 10 equally-spaced points spanning the whole foot are taken along the $x$-axis. The $\mathcal{Y}$-coordinates of these points, $\bar{y}_i$, $i=1,2,...,10$, which will serve to define the foot boundary through cubic splines, are the design variables of the optimization problem. The cost function to be minimized is the discrepancy between the histories of the ground reactions provided by the foot-ground contact model and those obtained from the inverse dynamic analysis. Normal and tangential forces, as well as the reaction moment, are considered into the cost function, scaling the reaction moment by a factor of 100 in order to balance the weight of the three components. The genetic algorithm $ga$ from Matlab has been used, for which no initial guess is required, and the resulting feet boundaries are depicted in Figure 3.

![Figure 3. Feet boundaries obtained from the optimization pre-process.](image)

3 CONTROL-BASED FORWARD DYNAMIC ANALYSIS

As said in the Introduction, the objective of this work is to perform the forward dynamic analysis of an acquired gait motion, by placing actuators at the joints only, thus leading to an underactuated system. Controllers governing the actuators are to track a number of outputs, which can be trajectories, forces or a combination of both. In what follows, a CTC-like approach for underactuated systems is described, which provides the inputs of the controllers as functions of the mentioned outputs.

The equations of motion of the underactuated system have been provided in (1), but are reproduced here for clarity,

$$M\ddot{z} = \mathbf{Q} + B\mathbf{u} \quad (4)$$

The required outputs are considered to be either functions of the coordinates (e.g. joint trajectories), or of the coordinates and their first derivatives (e.g. ground reactions produced by a force model).

$$\mathbf{y} = \begin{bmatrix} y_1(z) \\ y_2(z,\dot{z}) \end{bmatrix} \quad (5)$$

Differentiating (5) with respect to time (twice for $y_1$, and once for $y_2$), and substituting then $\dot{z}$ from (1) yields,
\[
\dot{y} = \begin{bmatrix} \dot{y}_1(z) \\ y_2(z, \dot{z}) \end{bmatrix} = \begin{bmatrix} H_{1z} & H_{2z} \\ H_{1z} & H_{2z} \end{bmatrix} \begin{bmatrix} \dot{z} \\ \dot{\dot{z}} \end{bmatrix} = \begin{bmatrix} H_{1z} \\ H_{2z} \end{bmatrix} \begin{bmatrix} \dot{z} \\ \dot{\dot{z}} \end{bmatrix} + \begin{bmatrix} H_{1z} \\ H_{2z} \end{bmatrix} \begin{bmatrix} \dot{z} \\ \dot{\dot{z}} \end{bmatrix} z = Az + D\dot{z} = Az + DM^{-1}(Q + Bu) \tag{6}
\]

so that the vector of actuations \(u\) can be worked out from Eq. (6) as,
\[
u = \left( DM^{-1}B \right)^{-1}\left( \dot{y} - Az - DM^{-1}Q \right) \tag{7}
\]

Now, calling \(P = DM^{-1}B\), and considering that feedback is introduced for the outputs, it results,
\[
u = P^{-1}\begin{bmatrix} \dot{y}_1 + C_v(\dot{y}_1 - \dot{y}_1) + C_p(y_1 - y_1) \\ \dot{y}_2 + K_p(y_2 - y_2) \end{bmatrix} - Az - DM^{-1}Q \tag{8}
\]

where the asterisk indicates the desired values of the outputs, different from the current ones (without asterisk), and \(C_v\), \(C_p\) and \(K_p\) are diagonal matrices containing the gains associated to each output.

If the number of outputs is equal to that of actuators, matrix \(P\) is square and the required inputs can be determined from (8). If the number of outputs is greater than that of actuators, the required outputs can be satisfied in a minimum squares sense only, the system of equations to be solved being,
\[
u = \left( P^T WP \right)^{-1} P^T W \begin{bmatrix} \dot{y}_1 + C_v(\dot{y}_1 - \dot{y}_1) + C_p(y_1 - y_1) \\ \dot{y}_2 + K_p(y_2 - y_2) \end{bmatrix} - Az - DM^{-1}Q \tag{9}
\]

with \(W\) the weight diagonal matrix which assigns more weight to more relevant outputs.

As the conventional CTC, this method is quite robust with respect to the gain values. In this work, \(C_p\) has been given the value \(10^3\), \(C_v\) the value corresponding to critical damping for the error dynamics \([10]\), i.e. \(C_v = 2\sqrt{C_p} = 2\sqrt{10^3} = 63.2456\), and \(K_p\) the value \(10^3\).

4 OUTPUT SELECTION

Different alternatives in the choice of the outputs have been investigated, looking for the one that yields the best agreement between the acquired motion and the result of the forward dynamic simulation. Here, the most representative options are described. Note that the human model considered has 14 degrees of freedom and 11 inputs (joint actuators).

The first strategy tested (case 1) was to define as many outputs that inputs, i.e. 11, choosing as outputs some 11 coordinates from the configuration vector \(z\) defined in (2). The selected outputs were the joint relative angles, while the three coordinates of the base body (trunk) were left free. The upper part of (8) was used to calculate the required inputs along the simulation.

The second strategy tested (case 2) was to define more outputs than inputs, choosing as outputs the 14 coordinates of the configuration vector \(z\) defined in (2). The upper part of (9) was used to calculate the required inputs along the simulation. In this case, the weights for the outputs must be decided, and used to build the weight matrix \(W\). The evolutionary optimization method known as Covariance Matrix Adaptation Evolution Strategy (CMA-ES) \([11]\), whose Matlab implementation has been downloaded from https://www.lri.fr/~hansen/, was applied to find the optimum values of the weights, which are gathered in Table 2 (rounded).
Table 2. Weights of the different outputs in case 2.

<table>
<thead>
<tr>
<th></th>
<th>x</th>
<th>y</th>
<th>α_1</th>
<th>α_2</th>
<th>α_3</th>
<th>α_4</th>
<th>α_5</th>
<th>α_6</th>
<th>α_7</th>
<th>α_8</th>
<th>α_9</th>
<th>α_10</th>
<th>α_11</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>98</td>
<td>98</td>
<td>99</td>
<td>8</td>
<td>40</td>
<td>0.01</td>
<td>2</td>
<td>27</td>
<td>16</td>
<td>30</td>
<td>2</td>
<td>42</td>
<td>2</td>
</tr>
</tbody>
</table>

The third strategy tested (case 3) was again to define more outputs than inputs, but choosing this time as outputs the 14 coordinates of the configuration vector \( z \) defined in (2) plus the three ground reaction components (normal and tangential forces and reaction moment) at each foot, i.e. 6 ground reaction components, leading to a total number of 20 outputs. The whole equation (9) was used to calculate the required inputs along the simulation. In this case, the weights required to build matrix \( W \) must be decided too: the kinematic outputs have been assigned a weight value 1, the ground normal reaction forces have been assigned a weight value \( 10^{-3} \), and both the ground tangential reaction forces and the reaction moments have been assigned a weight value of \( 10^{-2} \), so that the 20 outputs have a similar order of magnitude.

The agreement between the acquired motion (reference) and the result of the forward dynamic simulation (case 1, 2 or 3) is measured by means of the RMSE between the histories of the Cartesian coordinates of the black/white points in the human model shown in Figure 1 (right) corresponding to the reference and the case considered (1, 2 or 3). The obtained RMSE values are listed in Table 3.

Table 3. RMSE of the resulting motion with respect to the acquired motion.

<table>
<thead>
<tr>
<th>Case 1</th>
<th>Case 2</th>
<th>Case 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1070</td>
<td>0.0382</td>
<td>0.4818</td>
</tr>
</tbody>
</table>

It can be seen that the best correlation is obtained for case 2, i.e. when all the independent coordinates in the configuration vector \( z \) are selected as outputs, although in a minimum squares sense. Note also that, in this case, the weights of the different outputs have been optimized.

More detailed results are presented in the following. Figure 4 shows the histories of the three coordinates of the base body (trunk) for the three cases studied, and compares them with the result of the inverse dynamic analysis, taken as reference.

Figure 4. Coordinates of the base body (trunk) obtained with different control strategies vs reference.
Figure 5 plots the histories of the right hip angle, $\alpha$, for the three cases studied, and compares them with the result of the inverse dynamic analysis, taken as reference.

![Figure 5](image)

**Figure 5.** Right hip angle obtained with different control strategies vs reference.

Figure 6 gathers the histories of the normal ground reaction force at the left foot for the three cases studied, and compares them with the result of the inverse dynamic analysis, taken as reference. Note that the simulation starts with the heel-strike of the right foot.

![Figure 6](image)

**Figure 6.** Normal ground reaction force obtained with different control strategies vs reference.

In the last three figures, it can be seen that cases 1 and 2 provide good motion correlation, although some peaks can be observed in the ground reactions. Conversely, case 3 yields an
excellent correlation of the ground reactions, at the prize of being far from following the motion. As said before, weighting factors have been assigned in this case so as to provide equivalent relevance to all the outputs. Perhaps reducing the weight of ground reactions could lead to a better fitting of the motion.

To provide a clearer illustration of the obtained gaits, Figure 7 compares the resulting model motion with the acquired motion in the second case (best correlation).

As it can be seen in Figure 7, the main source of discrepancy in case 2 is the sliding between foot and ground during the support phase, due to the tangential force model. A stick-slip model could help to improve this aspect.

5 CONCLUSIONS

An acquired gait motion has been analyzed by a control-based forward dynamic approach, considering actuation at the joints only, so that the resulting system is underactuated. A CTC-like method has been applied to obtain the control inputs. Three cases have been tested: selecting as outputs as many system degrees of freedom as inputs; selecting as outputs all the system degrees of freedom in a minimum squares sense; selecting as outputs all the system degrees of freedom plus the ground reactions in both feet, again in a minimum squares sense.

First of all, it has been demonstrated that the definition of feet boundaries which are in good agreement with the ground reactions is essential to preserve gait stability during the simulation.

Regarding the three cases tested, the obtained results show, on the one hand, that tracking the motion through the proposed control-based forward dynamic approach is possible, despite the unstable character of gait. On the other hand, it is observed that imposing the motion of all the system degrees of freedom, yet in a minimum squares sense (case 2), works better than exactly imposing the motion of as many degrees of freedom as actuators (case 1). Moreover, the former strategy is quite robust with respect to the values of the weighting factors. If, besides the motion of all the system degrees of freedom, the ground reactions are also included as outputs (case 3), the correlation deteriorates.

As described in the paper, only a limited number of tests have been carried out. Therefore, the conclusions extracted here cannot be considered definitive, since many options can still be explored within the implemented strategies. For example, in case 1, a criterion could be applied consisting of selecting as outputs those coordinates corresponding to the largest pivots of matrix $P$, since the remaining coordinates are those with the highest capacity to control the system motion through their variation. In case 3, instead of assigning equal weight values to all the outputs, optimized sets of weighting values could be sought.
ACKNOWLEDGMENT

The support of this work by the Spanish Ministry of Economy and Competitiveness (MINECO) under project DPI2012-38331-C03-01, cofinanced by the European Union through EFRD funds, is greatly acknowledged.

REFERENCES


A new Model Based Estimation Algorithm for Train Axle Counting and Detection

B. Allotta, P. D’Adamio, A. Innocenti, E. Meli L. Pugi

Department of Industrial Engineering
University of Florence
Via Santa Marta n°3, 50139, Florence
benedetto.allotta@unifi.it pierluca.dadamio@unifi.it
alice.innocenti@unifi.it enrico.meli@unifi.it
luca.pugi@unifi.it

ABSTRACT
Train detection is an important research topic in the railway field affecting both vehicles and line safety. Currently, the European Train Control System ETCS (Level 1 and 2) provides the train localization functionalities by using track circuits and/or axle counter systems: the problem of these solutions is represented by the reliability and high cost of track circuit and axle counter installation and of the related equipment management. This paper presents an innovative train detection algorithm, able to perform the train localization and, at the same time, to estimate its speed, the crossing times on a fixed point of the track and the axle number. The algorithm can manage different types of input, measured on the track: more particularly, all the inputs are processed through cross-correlation operations to extract the required information. A suitable and accurate multibody model of railway vehicle and flexible track has been also developed by the authors to test the algorithm when experimental data are not available and in general, under any operating conditions (fundamental to verify the algorithm accuracy and robustness). The railway vehicle chosen as benchmark is the Manchester Wagon, modelled in the Adams VI-Rail environment. The physical model of the flexible track has been implemented in the Matlab and Comsol Multiphysics environments. A simulation campaign has been performed to verify the performance and the robustness of the proposed algorithm, and the results are quite promising. The research has been carried out in cooperation with Ansaldo STS and ECM Spa.

Keywords: Train detection, Multibody models of railways vehicles, Flexible tracks.

1 INTRODUCTION
With the increase of vehicle speed and traffic in the modern railways, a robust signalling system is fundamental to ensure the safety and reliable railway services [1]. In particular, the main safety properties of a reliable signalling system are the train detection, the railway traffic monitoring and the speed control on the train in the rear, to manage its movement according to the position of the train in front [2, 3]. Track circuits [4, 5] and axle counters [6, 7, 8] have been widely used to implement the train detection phase. Both solutions have disadvantages in terms of high installation cost and invasiveness for railway track (in the case of track circuits) and EMI interference (in the case of axle counters based on the electromagnetic technology). For this reason, many recent research works in literature propose a new simple sensor technology, the fiber Bragg grating sensor [9, 10], to be free from the EMI interference problem and to enhance the accuracy and reliability of the train detection of the signalling system [11, 12]. Thanks to this innovative technology, the complexity of the measure system has moved from the sensor part to that of peak reading in the received signal [13, 14, 15]. These works are really important but are solutions customized on specific experimental data that do not offer a general analysis approach and do not meet the question of a robustness analysis against different signal-to-noise ratio of the input signal.

In this scenario, with the goal of obtaining a more general and reliable method, the studied work is collocated. The proposed train detection algorithm aims at providing the localization of the train,
in terms of speed, crossing times on a fixed point of the track and axles number estimation. The formulation of the algorithm is quite general and it can be customized for several track measurement inputs (vertical loads on sleepers, stresses, strains, etc). Consequently it can be employed in different typologies of measurement stations and measurement chains: in the work case the chosen input is the vertical load on the sleeper [16].

Starting from the input signal, the estimation algorithm uses the cross-correlation to compute the crossing times of the train axles on the sleeper and consequently the vehicle speed; then, an operation of maximum peak detection, computed on the autocorrelations signal, followed by a time filtering based on a cut-off threshold, has been implemented to implement the axle counter function. All the post-processing operations are based only on auto and cross-correlation techniques and this represents a point of novelty for the proposed method, because it increases the simplicity and the reliability of the system. The algorithm is a low expensive and invasive method suitable for the Bragg fiber technology and is a robust solution against external noise and disturbances. The proposed approach turns out to be robust because, being based on correlation operations, it considers the global shape of the input signal, differently from other algorithms present in the literature that make use only of the local peaks of the measured signal.

A suitable multibody model of the railway vehicle (developed in Adams VI-Rail by the authors) and of the flexible track (developed in Matlab and Comsol) has been also developed to test the algorithm when experimental data are not available [17]. These two models interact online through a global contact model, developed by the authors in previous works [18, 19]. The considered railway vehicle is the Manchester Wagon [20]. The physical model of the system has been partially validated in the past by means of experimental data provided by Ansaldo STS [17]. These models are fundamental to test the algorithms in terms of accuracy and robustness under any operating conditions, also the most stressful ones. A series of simulation campaigns has been made to test the algorithm performance by varying the vehicle speed, the weights, the noise level on the input signal and the results are quite promising. The research has been carried out in cooperation with Ansaldo STS and ECM S.p.a..

2 GENERAL ARCHITECTURE OF THE SYSTEM

The architecture of the train detection system (Fig.1) is composed of two parts: the physical model and the estimation algorithm. The input of the estimation model can be classified into two types: experimental data measured on the real railway track or, in absence of them, data provided by a physical model. The purpose of this arrangement consists in the possibility of testing the algorithm performance even when experimental data are not available, that is fundamental to test the algorithms under any operating conditions (not possible with experimental data).

The physical model of the railway track consists of two sub-systems (see Fig. 2):

- 3D multibody model of the vehicle (in the studied case the Manchester Wagon [20]), implemented with VI-Rail software;
- 3D finite-element model (FEM) of the flexible railway track, developed in Comsol environment.

This two models interact online through a global wheel-rail contact model, developed by the authors in previous works [18][19].

At each time integration step, the multibody model evaluates the kinematic variables (position, orientation and their derivatives) of each wheel; at the same time, the finite-element model (FEM) of the railway track evaluates the position, orientation and their derivatives for each node of the beam that represents the rail. Both the kinematic variables are then sent as inputs to the global contact model, that returns the global contact forces to be applied to the wheel and the rail.

Once the simulation is finished and the vertical forces on the sleepers $F_{r/i}$, with $i=1..N_{st}$ ($N_{st}$ is the
minimum \( i \) of measurement sleepers included in the measurement station (indicating the vertical forces on the \( i \)-th sleeper measured at the right or left side of the train) are obtained, the estimation part begins. It is composed of two phases (both implemented in Matlab): the first one computes the auto correlation of each input signal coming from the sleepers \( F_{zr/l}^{i} \) and the cross-correlation among each pair of input signals \( F_{zr/l}^{i}, F_{zr/l}^{j} \). The second phase, instead, aims at elaborating the signals previously obtained to determine the vehicle parameters as speed \( V \), crossing times \( t_{i} \) on the \( i \)-th sleeper \( x_{i} \) and finally the number of train axles \( N \). The signal processing operations used in the second phase will be described in detail in Chap. 3.

3 TRAIN DETECTION ALGORITHM

The train detection algorithm aims at determining different train parameters like the crossing times of vehicle and wheelset on the sleepers, vehicle speed and axles number. The novelty of the proposed estimation method is that all these train quantities can be computed by means of only auto and cross-correlation operations. The used track inputs are the signals coming from the force sensors located on the left and right side on every sleeper \( F_{zr/l}^{i}, F_{zr/l}^{j} \) (see Fig. 5), even if the algorithm can manage different signal input (i.e. the vertical forces acting on the sleepers, the rail forces on the sleepers measured at the right or left side of the train) are obtained, the estimation part begins. It is composed of two phases (both implemented in Matlab): the first one computes the auto correlation of each input signal coming from the sleepers \( F_{zr/l}^{i} \) and the cross-correlation among each pair of input signals \( F_{zr/l}^{i}, F_{zr/l}^{j} \). The second phase, instead, aims at elaborating the signals previously obtained to determine the vehicle parameters as speed \( V \), crossing times \( t_{i} \) on the \( i \)-th sleeper \( x_{i} \) and finally the number of train axles \( N \). The signal processing operations used in the second phase will be described in detail in Chap. 3.

3 TRAIN DETECTION ALGORITHM

The train detection algorithm aims at determining different train parameters like the crossing times of vehicle and wheelset on the sleepers, vehicle speed and axles number. The novelty of the proposed estimation method is that all these train quantities can be computed by means of only auto and cross-correlation operations. The used track inputs are the signals coming from the force sensors located on the left and right side on every sleeper \( F_{zr/l}^{i}, F_{zr/l}^{j} \) (see Fig. 5), even if the algorithm can manage different signal input (i.e. the vertical forces acting on the sleepers, the rail
shear and bending, longitudinal strain and stress on rail). Fig. 3 illustrates the right and left vertical forces acting on the first sleeper (vehicle speed $V = 20\, \text{m s}^{-1}$ and car body mass $M = 40\, \text{t}$) of the measurement station: there are four peaks related to the four axles of the Manchester Wagon.

![Figure 3](image_url)

Figure 3. (left) Vertical forces $F_{zr}^1$, $F_{zl}^1$ acting on the right and left side of the first sleeper of the measurement station (performed at a vehicle speed $V=20\, \text{m s}^{-1}$) and a car body mass $M=40\, \text{t}$, (right) Original input signal $F_z^1$ (red line) and input signal with added noise (green line): Signal-to Noise ratio (SNR)=12dB performed at a vehicle speed $V=20\, \text{m s}^{-1}$ and a car body mass $M=40\, \text{t}$.

In order to accurately reproduce the signal acquisition, different noise levels have been added to the input signal. For example, Fig. 3 represents two signals (with or without the added noise), in which a noise has been added on the input signal to obtain a signal input-to-noise ratio equal to 12 dB (vehicle speed $V = 20\, \text{m s}^{-1}$ and car body mass $M = 40\, \text{t}$). The considered signal is the mean between the left and right force input signals $F_z^i = \frac{F_{zr}^i + F_{zl}^i}{2}$, in order to reject possible disturbances due to the lateral motion of the vehicle. The first step consists, firstly, in performing the autocorrelation on each obtained signals $F_z^i$ and then in the cross correlation between all the possible pairs of signals $F_z^i$ and $F_z^j$. The generic expression of the cross-correlation between two signals $(i,j)$ is:

$$R_{ij}(m) = \sum_{n=0}^{N-m-1} F_z^i(m+n) \, \overline{F_z^j(n)} \tag{1}$$

digitalized with N samples (m indicates the m-th sample of the correlation signals), where $\overline{F_z^j(n)}$ is the complex conjugate of the discretized signal $j$ and $F_z^i(m+n)$ indicates the discretized signal $i$, shifted of m samples. $R_{ii}(m)$ indicates the auto correlation signal. By means of the correlation operations, it is possible to evaluate the degree of true similarity between all the pairs of signals. In the second step, the attention is focused on the samples corresponding to the maximum value of the cross correlation signals: starting from the difference between the sample corresponding to the maximum value of the autocorrelation of the signal $F_z^i$ and the one corresponding to its cross correlation with the $F_z^j$ signal, it is possible to compute the time delay between the $F_z^i$, $F_z^j$ signals just multiplying this difference by the sample time integration step $\Delta t$. Through this method, the time shifts between all the pairs of input signals can be easily determined. Once known the time delays, the vehicle speed can be computed just dividing the distance between the corresponding sleepers by the time shift previously found for the signals. An example with two sleepers located...
at $X_i$ and $X_j$ positions, spaced of $d_{ij}$ apart, is reported:

$$m_i = \text{argmax} \ R_{ii}(m) \quad m_{ij} = \text{argmax} \ R_{ij}(m)$$

$$\Delta T_{ij} = \Delta t |m_{ij} - m_i| = t_j - t_i \quad d_{ij} = X_j - X_i \quad V = \frac{d_{ij}}{\Delta T_{ij}}$$

where $m_i$ and $m_{ij}$ are the samples corresponding respectively to the maximum value of autocorrelation of $i$-th sleeper signal $R_{ii}$ and cross correlation $R_{ij}$ between the $i$-th and the $j$-th ones; $\Delta t$ is the sample time and $\Delta T_{ij}$ is the time shift between the $i$-th and the $j$-th force signals (corresponding to the two sleepers); $V$ represents the vehicle speed, computed dividing $d_{ij}$ by the corresponding time delay $\Delta T_{ij}$. Fig. 4 shows the case of time shift between the first and the tenth signals coming from the sleepers. Fig. 4 highlights also why the auto/cross correlation operations have been chosen to implement the estimation algorithm: the amplitude of the auto/cross correlation signal is bigger than that of the force signal coming directly from the sensor (see Fig. 3), and so it guarantees a major robustness against input noise and disturbances.

To compute the crossing times on sleepers it is sufficient to use the signal time shifts among the different sleepers, starting from the first one to the last one:

$$t_i = t_0 + \sum_{j=0}^{i-1} \Delta T_{j,j+1}.$$  

The method used for the axle number detection is still based on the correlation theory: also in this case it has been decided to work with the correlation signals rather than the direct signals coming from the force sensors because the correlation operation increases the signals of several orders of magnitude when there is a good degree of true similarity and so it guarantees a better robustness against the input noise and disturbances. In addition, it has been observed that the peaks of true similarity, composing the correlation signal, are deeply related to the peaks of the original signal, representative of the passage of the axles on the sleepers. Fig. 4 illustrates the auto correlation signal when one, two or three of the axle peaks are present in the force signal instead of four.

The number of significant autocorrelation peaks is 9 if 4 axles are present, while decreases to 5 if 3 axles are present, decreases to 3 if 2 axles are present and finally decreases to 1 if 1 axle is
present. Through this method, counting the autocorrelation peaks, it is possible to count the signal force peaks and hence the crossing train axles number.

Once the auto correlation signal is obtained, it has been limited inside a sample window, the size of which is determined to include only the significant samples. An algorithm to determine the signal local peaks is implemented, based on the signal $R_{i,i}$, on its derivative $R'_{i,i}$ and on suitable cut off thresholds to separate the signal peaks from the noise peaks [13].

4 PHYSICAL MODEL OF THE RAILWAY TRACK

In order to generate suitable simulation campaigns to test the proposed algorithm when experimental data are not provided and to test the algorithm under any operating conditions, a model involving all the components of the track and the vehicle is required. The physical model consists of a 3D finite element model (FEM) of the infrastructure (rail, sleepers and ballast), a 3D multibody model of the vehicle [21] and a contact model describing the interaction between the vehicle wheels and the rail [22, 23, 24]. The vehicle model and the infrastructure model interact online during the simulations by means of the 3D global contact model, specifically developed to improve reliability and accuracy of the wheel-rail contact points detection.

The whole model can provide as outputs different track measurement signals like vertical loads on the sleepers, stresses and strains on the rail etc. In this work the vertical forces on the sleepers $F_{zr}$ and $F_{zl}$ are considered as inputs of the train detection algorithm (see Chapter. 3); (see Fig. 5).

4.1 The vehicle model

The railway vehicle chosen for the dynamic simulations is the Manchester Wagon, the mechanical structure, elastic and damping characteristics and the inertia properties of which are easily available in literature [20]. The main general characteristics of the vehicle are referred to Tab. 1. The multibody 3D model of this vehicle [21] has been widely studied and validated in different conditions. The model of the Manchester Wagon consists of seven rigid bodies: one car body, two bogies and four wheelsets. As regards the suspension stages and the corresponding stiffness and dumping properties one can see the previous bibliography references.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bogie pivot distance</td>
<td>19 000  mm</td>
</tr>
<tr>
<td>Bogie wheelbase</td>
<td>2 560  mm</td>
</tr>
<tr>
<td>Wheel diameter</td>
<td>920     mm</td>
</tr>
<tr>
<td>Height above ToR level of bogie CoG</td>
<td>600     mm</td>
</tr>
<tr>
<td>Height above ToR level of coach CoG</td>
<td>1 800    mm</td>
</tr>
<tr>
<td>Longitudinal and lateral offset of coach CoG</td>
<td>0 mm</td>
</tr>
</tbody>
</table>

4.2 Measurement Layout

The train detection station can manage different measure layouts characterized by various measure points (few if possible to reduce both the measure station dimensions and the economic costs) distributed along the railway track on both the track sides. In the present research activity, the reference layout of the adopted measurement station consists of three measure points on both rail side. As represented in Fig. 5, the sensors are placed on three measurement sleepers ($X_i$) indicates the position of the i-th sleeper).
Furthermore, to test the robustness of the estimation algorithm, many measurement layouts have been considered, characterized by a different number of sleepers and different distances among them.

5 PERFORMANCE OF THE TRAIN DETECTION ALGORITHM IN ESTIMATING THE VEHICLE SPEED $V$ AND THE CROSSING TIMES $t_i$ ON THE MEASUREMENT SLEEPERS

This chapter describes the performance of the train detection algorithm in estimating the speed and the crossing time instants on sleepers starting from the knowledge, among all the possible measurements inputs of the vertical loads on the sleepers. The estimation algorithm has been tested through a simulation campaign, in which the attention is focused on the estimation behaviour as a function of vehicle speed $V$, car body mass $M$ and input signal-to-noise ratio (SNR). For the testing of the estimation algorithm performances, the reference measurement layout is the one reported in Fig. 5, composed of three force sensors located on the left and right side of three consecutive sleepers. In the following, as previously said, this layout will be modified so as to verify the robustness of the proposed estimation method. In Tab. 2 the considered ranges of the previous quantities are reported, where $N_V$, $N_M$, $N_{snr}$ represent respectively the number of simulated values of $V$, $M$ and signal to noise ratio SNR (the ratio between the power of the input signal and the one of noise input level).

![Figure 5. Measurement Layout- (position of the force sensors on the sleepers)](image)

Table 2. Variation ranges of $V$, $M$ and SNR adopted for the simulation campaign.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Min.</th>
<th>Max.</th>
<th>$N_V/N_M/N_{snr}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Velocity (m s$^{-1}$)</td>
<td>10</td>
<td>40</td>
<td>4</td>
</tr>
<tr>
<td>Car-body Mass (t)</td>
<td>10</td>
<td>50</td>
<td>5</td>
</tr>
<tr>
<td>Signal-to-noise ratio (dB)</td>
<td>5</td>
<td>12</td>
<td>4</td>
</tr>
</tbody>
</table>

The global performance of the algorithm in terms of vehicle speed and the crossing times on the sleepers are reported and studied by considering the relative error $e^{sim}(V, M, SNR)$ on 100 different runs repeated according to the Monte Carlo approach for each value of $M$ and $V$ (the maximum error value is considered):

$$
\begin{align*}
    e^v_{sim} &= \frac{|\hat{V}_{sim} - V|}{V} \\
    e^t_{sim} &= \frac{|\hat{t}_{i, sim} - t_i|}{t_i}
\end{align*}
$$

(5)

where $V$ and $t_i$ represents the nominal values of the speed and crossing times respectively, and $\hat{V}_{sim}$, $\hat{t}_{i, sim}$ indicate the estimated ones. All the results are obtained with a time integration step.
equal to $\delta T = 0.001$ s. Fig. 6 shows a comparison between the crossing times percentage errors, and their behaviour as a function of vehicle speed $V$ and car body mass $M$; each graph is related to a different value of the signal to noise ratio $\text{SNR}$ of the input signal (from 8 to 12 dB). In particular, for each SNR test case, the percentage errors on the second and third sleeper are reported (the first sleeper is assumed as reference and is coincident with a crossing time instant equal to zero).

Figure 6. Percentage relative error on the crossing times $t_i$ on the second (on the left) and third (on the right) sleeper as a function of nominal speed $V$ and car body mass $M$, for different values of $\text{SNR}$.

The maximum resulting errors in the simulation campaigns are equal to 0.58 % and to 0.45 % for the second and third sleeper respectively, (related to a simulation performed considering the following values: $V=30$ m/s$^{-1}$, $M=50$ t and $\text{SNR} = 5$ dB).

The algorithm performances in estimating the crossing times on the sleepers are very important because the speed estimation uses the same time delays (between every pair of measurement sleepers) to perform the speed calculation: therefore an estimation error on the time delays would deeply affect the estimation of the speed. The following figure (Fig. 7) shows a comparison among the speed percentage errors and their behaviour as a function of vehicle speed $V$ and car body mass $M$; each graph is related to a different value of the signal to noise ratio $\text{SNR}$ of the input signal (from
5 to 12 dB). The maximum error (equal to 0.54%) occurs in the case of the simulation performed considering the following values: \( V = 30 \, \text{m/s} \), \( M = 30 \, \text{t} \) and \( \text{SNR} = 5 \, \text{dB} \).

The obtained results highlight the good performance of the algorithm against different values of input signal to noise ratio and consequently the capability of the algorithm to deal with measurement acquisition affected by a high noise level. Furthermore, the increase of the vehicle speed and mass does not seem to influence the performance of the method. This is mainly due to the fact that, differently from other approaches present in the literature, the new algorithm exploits the global shape of the measurement signal, not only its local peaks.

### 6 Performance of the train detection algorithm as axles counter

The simulation campaign to test the algorithm capability as axles counter has been performed with variable vehicle parameters and input noise as indicated in Tab. 3.

#### Table 3. Variation ranges of \( V \), \( M \) and \( \text{SNR} \) adopted for the simulation campaign.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Min.</th>
<th>Max.</th>
<th>( N_V/N_M/N_{\text{SNR}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Velocity (m s(^{-1}))</td>
<td>10</td>
<td>40</td>
<td>4</td>
</tr>
<tr>
<td>Car-body Mass (t)</td>
<td>10</td>
<td>50</td>
<td>5</td>
</tr>
<tr>
<td>Signal-to-noise ratio (dB)</td>
<td>8</td>
<td>12</td>
<td>5</td>
</tr>
</tbody>
</table>

The performance of the algorithm in estimating the train axles number has been evaluated considering the relative percentage error defined as follows:

\[
e^{\text{sim}}_N = \frac{\left| \tilde{N}_{\text{tot}}^{\text{sim}} - N_{\text{tot}} \right|}{N_{\text{tot}}} \tag{6}
\]

where \( N_{\text{tot}} \) represents the number of crossing vehicle axles and \( \tilde{N}_{\text{tot}}^{\text{sim}} \) its estimated value. To test the axle counting performance, the results have been computed performing 100 runs of the algorithm, according to the Monte Carlo approach, for each value of vehicle speed \( V \) and car body mass \( M \) (the maximum error is considered). The used SNR values are those indicated in Tab. 3.

Results show the good performance of the estimation algorithm as axles counter, especially with a SNR bigger than 10 dB (see Tab. 4), in which the percentage errors are below than 3%. For
Table 4. Error on estimating the axle number with variable speed $V$, car body mass $M$ and SNR of the input signal.

<table>
<thead>
<tr>
<th>Speed [m s$^{-1}$]</th>
<th>Mass [t]</th>
<th>S/N=8db Err.[%]</th>
<th>S/N=9db Err.[%]</th>
<th>S/N=10db Err.[%]</th>
<th>S/N=11db Err.[%]</th>
<th>S/N=12db Err.[%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>$M=10t$</td>
<td>12</td>
<td>4</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$M=20t$</td>
<td>17</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$M=30t$</td>
<td>18</td>
<td>5</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$M=40t$</td>
<td>16</td>
<td>9</td>
<td>2</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$M=50t$</td>
<td>15</td>
<td>10</td>
<td>3</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>20</td>
<td>$M=10t$</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$M=20t$</td>
<td>5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$M=30t$</td>
<td>6</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$M=40t$</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$M=50t$</td>
<td>8</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>30</td>
<td>$M=10t$</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$M=20t$</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$M=30t$</td>
<td>3</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$M=40t$</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$M=50t$</td>
<td>6</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>40</td>
<td>$M=10t$</td>
<td>4</td>
<td>4</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$M=20t$</td>
<td>12</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>$M=30t$</td>
<td>10</td>
<td>5</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$M=40t$</td>
<td>4</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>$M=50t$</td>
<td>3</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Each car body mass $M$ and SNR of the input signal, the highest errors in the axle counting occur for the test case with speed $V$ equal to 10 m s$^{-1}$: this further demonstrates the good performance of the algorithm, especially with high speed. Also in this case, an important role is played by the capability of the method to estimate the vehicle parameters starting from the global shape of the input signal, instead of only the local peaks.

7 CONCLUSIONS

In this paper the authors presented an innovative train detection algorithm with the aim of estimating the railway vehicle speed, its crossing time instants on the sleepers and finally its axles number. The algorithm is based on the measurement of the vertical forces on sleepers $F_{zr}$ and $F_{zi}$ performed through force sensitive elements placed over the sleepers in the section corresponding to the rail baseplate/pads. The novelty consists in using correlation operations to estimate all the different parameters of the railway vehicle. Thanks to this feature, the algorithm is general and applicable to different measurement layouts and to different input signals measured on the track. Furthermore, the proposed method, if compared to other approaches present in the literature, is capable of exploiting the global shape of the input signal to estimate the vehicle parameters (speed $V$, crossing time instants on the sleeper $t_i$ and axles number $N$) instead of only the peak values. A simulation campaign has been made to test the algorithm performance in estimating the vehicle parameters as a function of its speed $V$ and car body mass $M$. In order to simulate different measurement operating conditions, the vehicle parameters have been estimated with a signal to noise ratio starting from 5dB up to 15 dB. Results highlight the good performance of the algorithm in performing the estimation of the crossing times on the sleeper, vehicle speed and axle number in all the SNR range. Concerning the future developments, the aim is to optimize the estimation algorithm as train axles counter, using lower SNR values and a variable composition of the vehicle.
REFERENCES


An Innovative Model-Based Dynamic Weigh in Motion System for Railway Vehicles

P. D’Adamio, A. Innocenti, L. Marini, E. Meli, L. Pugi, A. Rindi

Department of Industrial Engineering
University of Florence
Via Santa Marta 3, 50139, Florence
pierluca.dadamio@unifi.it alice.innocenti@unifi.it
lorenzo.marini@unifi.it enrico.meli@unifi.it
luca.pugi@unifi.it andrea.rindi@unifi.it

1 ABSTRACT
One of the most important issues in railway research is represented by the accurate estimation of the axle loads of railway vehicles. **Weigh in Motion** (WIM) devices are designed to measure loads with the vehicle in motion, making the weighing process more efficient.

This paper is focused on an innovative algorithm for high speed WIM applications able to estimate the wheel loads of trains by means of indirect track measurements. The novelty of the proposed estimation method is its generality, the possibility to be used with different layouts and its robustness against numerical and measure noise. The main estimation procedure is based on least square (LSQ) minimization techniques, used to process the set of experimental physical input.

The whole WIM architecture has been developed in cooperation with Ansaldo STS and ECM SpA.

**Keywords:** Weigh-in-motion systems, Multibody modelling of railway vehicles, Flexible multibody modelling of railway track.

2 INTRODUCTION
In the last decades, the railway research proposed the study of different WIM **Weigh in Motion** systems for several applications as bridge-weigh in motion technology to determine train physical quantities (vehicle speed, axle distances and static axle loads ([1])), and systems for structural health monitoring (by using FBG sensor array ([2], [3]), or vibration sensor [4], [5]). To carry out the dynamical weighing, different measurement stations can be used, without stopping the railway traffic: both the railway industry and the scientific community provide accurate and reliable measurement systems [6, 7, 8, 9, 10]. In the present paper the authors present an innovative algorithm for the wheel or axle loads estimation, whose novelty is the capability to be used with different input quantities as vertical forces on the sleepers or deformations, stresses of the rail and with different measurement station layouts. In this paper, force sensors between wheels and sleepers are considered, because they do not cause invasive intervention on the rail and allow a mechanic protection from bad weather condition. Starting from the knowledge of indirect rail measurement of the vertical sleeper forces, the vertical load of each wheel is estimated, assuming that the effects of the single wheel loads on the track are superimposable (quasi linearity hypothesis). In this way the track response can be expressed as a weighted combination of the effects caused by a series of single nominal loads, moving along the track. Least square (LSQ) minimisation techniques [11] are used for the estimation phase, starting from the set of physical quantities selected as track inputs. Once the estimated vertical loads acting on the wheels are known, the algorithm is also able to estimate the longitudinal \( X_G \) and lateral \( Y_G \) coordinates of the centers of mass of the vehicles.

A further point of novelty of the proposed estimation method is represented by its capability of managing both real experimental data or simulated data: the last ones are provided by a physical model of the whole railway system. Numerical simulations are made to test the performances of the proposed algorithm, when experimental data are not available. In the developed model, both
Figure 1. General architecture of WIM system

physical and measurement noise have been added. In order to test the algorithm robustness, its performances with perturbed track parameters (stiffness and damping of sleepers and ballast) have been tested. The whole WIM architecture has been developed in cooperation with Ansaldo STS and ECM SpA.

3 GENERAL ARCHITECTURE OF THE SYSTEM

The general architecture of the developed WIM algorithm [9] is illustrated in Fig. 1 and it consists of two main parts: the physical model and the estimation algorithm. The aim of this arrangement consists in the chance of testing the algorithm performance also when experimental data are not available: the necessary track inputs are provided by the dedicated mechanical model through numerical simulations of a completely known vehicle transiting on the track. Such simulations are performed to obtain the dynamic response of the track, used in the estimation process. More precisely, the physical model is composed by two sub-models: the multibody model of the investigated vehicle (implemented in Adams VI-Rail environment) and the finite element model of the track (developed in Comsol environment) that, during the dynamic simulation, interacts online through a global contact model developed and validated by the authors in previous works [12, 13]. At each time integration step the multibody model of the vehicle evaluates the kinematic variables (position, orientation and their derivatives) relative to the wheelset and consequently to each wheel. The finite element track model evaluates the kinematic variables (position, orientation and their derivatives) of each rail. Rail and wheel kinematic variables are sent as inputs to the global contact model that calculates the global contact forces and sends these values back both to the vehicle multibody model and to the finite element track model. The estimation part is composed by the innovative algorithm (implemented in Matlab) and the module for the basis functions evaluation (developed in Comsol) described in chapter 5. The algorithm requires some additional information concerning the vehicle speed $V$, the axle number $n_{tot}$ and positions along the railway vehicle $x_{ai}$ with $i = 1, \ldots, n_{tot}$, measured using additional sensors or transmitted by the vehicle by means of low cost technologies. In this work the algorithm is based on the measurement of the vertical forces acting on the sleepers performed by means of force sensitive elements placed over the sleepers in the section corresponding to the rail baseplate/pads. These forces (simulated $F_{zn}^{f}$, $F_{zr}^{f}$ if provided by a physical model of the railway track or real $F_{zn}^{p}$, $F_{zr}^{p}$ if coming from experimental data), represent the physical track inputs of the WIM algorithm that, starting from the knowledge of these quantities, estimates the wheel or axle loads $\hat{N}$, longitudinal $\hat{X}_G$ and lateral $\hat{Y}_G$ position of the center of mass $G$ of the investigated vehicle, through suitable estimation procedures derived from the least squares minimization [14][15].
4 PHYSICAL MODEL OF THE RAILWAY TRACK

The physical model consists of three elements: a 3D finite element model of the infrastructure (rail, sleepers and ballast), a 3D multibody model of the vehicle and an innovative 3D wheel-rail contact model (Fig. 2). In the rest of the paper \( x_{ai} \) will denote the initial position of the \( i-th \) axle of the vehicle (the total number of the axles is \( n_{tot} \)), while the generic vertical right and wheel loads are indicated as \( N_{Ri} \) and \( N_{Li} \). The corresponding estimated wheel loads \( \hat{N}_{Ri} \) and \( \hat{N}_{Li} \) will be computed by the presented WIM algorithm; the weights of the wheelsets are included in the loads \( \hat{N}_{Ri} \) and \( \hat{N}_{Li} \). Rails are modelled as 6 degrees of freedom 3D beams, connected through visco-elastic elements to \( n_{sl} \) 2D rigid bodies representing rail sleepers, which are in turn supported by a visco-elastic foundation including the ballast properties. The visco-elastic elements are composed by lateral \( k_{ysl} \), vertical \( k_{zsl} \) and rotational \( k_{\phi sl} \) stiffness and lateral \( c_{ysl} \), vertical \( c_{zsl} \) and rotational \( c_{\phi sl} \) damping properties. The generic 2D sleeper is supported by a flexible foundation characterising the behaviour of the ballast through the lateral \( k_{ybal} \), vertical \( k_{zbal} \) and rotational \( k_{\phi bal} \) stiffness values and lateral \( c_{ybal} \), vertical \( c_{zbal} \) and rotational \( c_{\phi bal} \) damping values. The 3DOF system modelling the sleepers-ballast ensemble is described by the lateral \( y_{sl} \) and vertical \( z_{sl} \) translations and the rotation \( \phi_{sl} \) around the \( x_{sl} \) axis of the sleeper reference system. More details on the modelling and on the parameters of the rail-sleeper-ballast ensemble can be found in [16].

The investigated vehicle chosen for the dynamic simulations is the Manchester Wagon whose mechanical structure and elastic and damping characteristics are easily available in literature [17, 16]. The vehicle is composed of the car body, two bogies and four wheelsets. For further detail on the multibody 3D model of this vehicle one can refers to [16]. The vehicle and the infrastructure model interact online during the simulations by means of a 3D global contact model, specifically developed to improve reliability and accuracy of the contact points detection. In particular the adopted contact model is based on a two step procedure; the contact points detection [12, 13] and the global contact forces evaluation [18].

All the measurement systems have a limited operative band and for this reason the frequency...
effects on the generic considered input signals $T_r, T_l$ are introduced. These effects have been modelled with a second order low pass filter, applied to the physical signals $T_{rk}^{sim}, T_{lk}^{sim}$ relative to the measure points $x_{rk}$ and $x_{lk}$ (with subscripts r and l make reference to the respectively right and left measurement point) of the measurement station: $T_{rk}^{f}(t) = B_{2,ω_0}(s)T_{rk}^{sim}(t)$ and $T_{lk}^{f}(t) = B_{2,ω_0}(s)T_{lk}^{sim}(t)$ where $B_{2,ω_0}(s)$ is the second order Butterworth filter and $ω_n = 2πf_n$ is the cut frequency ($ω_0$ in rad/s and $f_n$ in Hz). Besides the frequency effects, also numerical noise and bias errors on the signal $T_{rk}^{f}, T_{lk}^{f}$ have been modelled: $T_{rk}^{fn}(t) = T_{rk}^{f}(t) + UT_{rk}^{T_l}μ_r, δ_{r/2}$ and $T_{lk}^{fn}(t) = T_{lk}^{f}(t) + UT_{rk}^{T_l}μ_l, δ_{l/2}$ where $μ_r, μ_l$ and $δ_{r/2}, δ_{l/2}$ are the means and the standard deviations of the disturbance distributions $U_T, U_{T_l}$.

5 WIM ALGORITHM

5.1 Vertical wheel loads estimation

The WIM algorithm (see Fig. 3) can manage both the generic simulated ($T_{rk}^{f/n}$ and $T_{lk}^{f/n}$) and experimental ($T_{rk}^{fp}$ and $T_{lk}^{fp}$) input data. The developed algorithm estimates the vertical right $N_{Rk}$ and left $N_{Lk}$ wheel loads knowing the specific track measurements chosen as input signals $T_{rk}$ and $T_{lk}$ measured respectively at $x_{rk}$ and $x_{lk}$ (indicating the abscissas of the right and of the left side of the $k-th$ measurement point with $k = 1, ..., N_m$, $N_m$ is the number of measurement points). The main hypothesis on which the proposed algorithm is base is that the effect of the generic load $N_{Rk}$ and $N_{Lk}$ on the generic track measurement input $T_{rk}$ and $T_{lk}$ is assumed to be no influenced from the presence of other loads (especially the contiguous ones). Evidently the linearity hypothesis (LH) must hold within the total range of velocities $V$ and cut frequencies $f_n$ considered for the studied systems. Then, the application of the superposition principle allows to estimate the track inputs $T_{rk}$ and $T_{lk}$ produced by the train to be estimated. According to the QL hypothesis, the track inputs $T_{rk}$ and $T_{lk}$ are respectively estimated through a linear combination of $2n_{tot}$ track fictitious input signals (namely the basis functions) produced by $2n_{tot}$ single fictitious loads $N_f$ (one for each vehicle wheel) shifted in the time of a delay $t_i$. The fictitious load $N_f$ must include the weight of the wheel itself. In this case the linear combination coefficients are equal to $\hat{N}_{Rk}/N_f$ and $\hat{N}_{Lk}/N_f$. A Least Squares Optimization (LQSO) is needed to minimize the approximation error and to optimize the values of $\hat{N}_{Rk}$ and $\hat{N}_{Lk}$.

The quantities $B_{Rk}^{f/n}$ and $B_{Lk}^{f/n}$ represent the chosen track fictitious response due to the transit of the $i-th$ fictitious load respectively on the right or on the left (denoted respectively with subscripts R and L) rail, measured at the right (r) side of the $k-th$ measurement point. Analogously, $B_{Rk}^{fp}$ and $B_{Lk}^{fp}$ indicate the chosen track responses due to the transit of the $i-th$ fictitious load respectively on the right or on the left rail, measured at the left (l) side of the $k-th$ measurement station.

The model of the rail infrastructure used in the fictitious system to evaluate the basis functions is similar to the one used to simulate the real physical model (see chapter 4) but, at the same time, it is quite simpler (only the vertical DOFs of rails, sleepers and ballast are considered). In real applications, the exact physical model is unknown and only a simplified approximate model can be used inside the WIM algorithm. Moreover, since the WIM procedure has to be fast and implemented almost in real-time, the fictitious model needed to compute the basis functions has to be necessarily simple.

The WIM algorithm estimation procedure will be described considering general track inputs $T_{rk}$ and $T_{lk}$. The right $T_{rk}$ and left $T_{lk}$ track inputs measured at the $k-th$ measurement point will be expressed as $T_{rk}(t) = T_{rk}(x_{rk}, t)$, $T_{lk}(t) = T_{lk}(x_{lk}, t)$ with $t \in [T_1, T_F]$. The position of a generic fictitious load $N_f$ along the track is defined as $x_f = x_{af} + t * V$ (where $x_{af} = 0 m$). The $2n_{tot}$ right-side fictitious track inputs $B_{Rk}^{f/n}$ and $B_{Rk}^{fp}$ and the $2n_{tot}$ left-side fictitious track inputs $B_{Lk}^{f/n}$ and $B_{Lk}^{fp}$ (in the present case the vertical forces acting on the sleepers) produced by $2n_{tot}$ single fictitious loads can easily be assessed by introducing suitable time delays $t_i = (x_{ai} - x_{af})/V$ and by applying such delays to the track responses to the transit of a single fictitious
load (i.e. single wheel transit): \( B_{rk}^k(t) = B_{rk}^k(t + t_i) \), \( B_{lk}^k(t) = B_{lk}^k(t + t_i) \), \( B_{rk}^l(t) = B_{rk}^l(t + t_i) \) and \( B_{lk}^l(t) = B_{lk}^l(t + t_i) \). The track inputs \( T_{rk} \) \( T_{lk} \) (simulated \( T_{rk}^{fn} \) \( T_{lk}^{fn} \)) and the experimental ones \( T_{rk}^{fp} \) \( T_{lk}^{fp} \) produced by the transit of the entire train can be approximated as follows:

\[
CT_{rk}(t) \approx T_{rk\,app}(t) = \sum_{i=1}^{n_{str}} B_{rk}^k \alpha_{ri} + \sum_{i=1}^{n_{str}} B_{lk}^k \alpha_{li} \quad (1)
\]

\[
T_{lk}(t) \approx T_{lk\,app}(t) = \sum_{i=1}^{n_{str}} B_{rk}^l \alpha_{ri} + \sum_{i=1}^{n_{str}} B_{lk}^l \alpha_{li} \quad (2)
\]

where a direct proportionality between the linear combination coefficients \( \alpha_{ri} \) \( \alpha_{li} \), the estimated vertical loads \( \hat{\bar{N}}_{ri} \) \( \hat{\bar{N}}_{li} \) and the fictitious vertical load \( \bar{N}_f \) holds: \( \alpha_{ri} = \hat{\bar{N}}_{ri}/\bar{N}_f \) and \( \alpha_{li} = \hat{\bar{N}}_{li}/\bar{N}_f \).

The time domain \( t \in [T_f, T_f] \) has been discretized with a sample time \( \Delta t \) equal to 0.001 s. Therefore, the \( T_{rk} \) \( T_{lk} \) track inputs are known only at the times \( t_h \) with \( h = 1,2,...,n_t \) \( n_t \) is the samples number while \( t_1 = T_f \) and \( t_{N_t} = T_f \). The same time discretization holds also for the fictitious track outputs \( B_{rk}^{k\,sp}, B_{lk}^{k\,sp}, B_{rk}^{l\,sp}, B_{lk}^{l\,sp} \) employed to estimate \( T_{rk}^{sp}, T_{lk}^{sp} \).

In matrix form, Equation 2 and 1 are: \( T_{rk} \approx B_{rk}^k \alpha_R + B_{lk}^k \alpha_L \) and \( T_{lk} \approx B_{rk}^l \alpha_R + B_{lk}^l \alpha_L \) where \( k = 1,\ldots,N_m, T_{rk}, T_{lk} \in \mathbb{R}^{n_x \times 1} \), \( B_{rk}^k, B_{lk}^k, B_{rk}^l, B_{lk}^l \in \mathbb{R}^{n_x \times n_{out}} \) and \( \alpha_R, \alpha_L \in \mathbb{R}^{n_{out} \times 1} \). Considering then the \( n_m \) measuring points, a more compact problem formulation can be obtained: \( T_f \approx B_{rk}^k \alpha_R + B_{lk}^k \alpha_L \) and \( T_f \approx B_{rk}^l \alpha_R + B_{lk}^l \alpha_L \), where \( T_f, T_f \in \mathbb{R}^{m_{out} \times 1} \), \( B_{rk}^k, B_{lk}^k, B_{rk}^l, B_{lk}^l \in \mathbb{R}^{m_{out} \times n_{out}} \) and \( \alpha_R, \alpha_L \in \mathbb{R}^{n_{out} \times 1} \). The following expression finally holds:

\[
\begin{bmatrix} T_f \\ T_f \end{bmatrix} \approx \begin{bmatrix} B_{rk}^k \\ B_{lk}^k \\ B_{rk}^l \\ B_{lk}^l \end{bmatrix} \begin{bmatrix} \alpha_R \\ \alpha_L \end{bmatrix} \quad (3)
\]

or, more briefly, \( T \approx B \alpha \) where \( T \in \mathbb{R}^{2m_{out} \times 1} \), \( B \in \mathbb{R}^{2m_{out} \times n_{out}} \) and \( \alpha \in \mathbb{R}^{2n_{out} \times 1} \).
In order to minimize the approximation error in Equation 3, \( T_{rk}, T_{lk} \) and \( T_{rk\, app}, T_{lk\, app} \) and, at the same time, to optimize the values of \( \hat{N}_{Ri}, \hat{N}_{Li} \), linear not-weighted least squares have been considered [14][15][11].

In the present research activity the vertical forces acting on the sleepers (denoted with \( F_{z\, rk}^n \) and \( F_{z\, lk}^n \)) have been adopted as track inputs. For the simulated vertical forces acting on the sleepers \( F_{z\, rk}^n \) and \( F_{z\, lk}^n \), equation (3) becomes: \( F_{zn} = B\alpha_{\text{sim}}. \)

By means of a least squares optimization (LQSO) (in this case linear and not-weighted), it is possible to minimize the squared 2-norms \( E_{fn}^2 = \|E_{fn}\|^2 \) of the approximation errors \( E_{fn} = B\alpha_{\text{sim}} - F_{zn}. \) This leads to \( \alpha_{\text{sim}} = (B^T B)^{-1} B^T F_{zn} \) where the matrix \( B^T B \) is invertible if and only if the rank of \( B \) is maximum. Starting from the knowledge of \( \alpha_{\text{sim}} \), it is possible to evaluate the values of the estimated vertical loads \( \hat{N}_{Ri, \text{sim}}, \hat{N}_{Li, \text{sim}} \):

\[
L_{CL}\alpha_{\text{sim}} = \hat{N}_{\text{sim}} / N_f \text{ where } \hat{N}_{\text{sim}} = \begin{bmatrix} \hat{N}_{R1}^\text{sim} \\ \hat{N}_{R2}^\text{sim} \\ \hat{N}_{R3}^\text{sim} \\ \hat{N}_{R4}^\text{sim} \\ \hat{N}_{L1}^\text{sim} \\ \hat{N}_{L2}^\text{sim} \\ \hat{N}_{L3}^\text{sim} \\ \hat{N}_{L4}^\text{sim} \end{bmatrix}^T \]
\]

with

\[
\begin{align*}
C_{\text{NN}}\hat{N}_R &= \begin{bmatrix} \hat{N}_{R1}^\text{sim} \\ \hat{N}_{R2}^\text{sim} \\ \hat{N}_{R3}^\text{sim} \\ \hat{N}_{R4}^\text{sim} \end{bmatrix} \\
\hat{N}_L &= \begin{bmatrix} \hat{N}_{L1}^\text{sim} \\ \hat{N}_{L2}^\text{sim} \\ \hat{N}_{L3}^\text{sim} \\ \hat{N}_{L4}^\text{sim} \end{bmatrix}^T.
\end{align*}
\]

5.2 Center of gravity estimation

The innovative WIM algorithm can estimate the lateral \( Y_G \) and \( X_G \) longitudinal coordinates of the vehicle center of gravity (see for example Fig. 4 where a two-bogies four-axles vehicle is schematically illustrated), starting from the knowledge of the estimated wheel loads \( \hat{N}_{Ri} \) and \( \hat{N}_{Li} \). Considering the horizontal plane containing the center of gravity of the railway vehicle and introducing the reference system shown in Fig.4 (the origin \( O \) coincides with the geometric center of the carbody), the moment equilibrium around \( X_B - \text{axis} \) and \( Y_B - \text{axis} \) can be respectively expressed as \( \sum_{i=1}^{n_{tot}}(b_R \hat{N}_{Ri} + b_L \hat{N}_{Li}) = 0 \) and \( \sum_{i=1}^{n_{tot}} a_i(\hat{N}_{Ri} + \hat{N}_{Li}) = 0. \)

Figure 4. Estimation of lateral and longitudinal coordinates of the vehicle center of gravity.

Using the nominal values of the geometrical quantities of the vehicle such as the longitudinal position inside the train of each axle \( x_{ai} \) and the distance between the nominal rolling radius (axle track) \( s \), the coefficients \( b_R, b_L \) and \( a_i \) can be re-written as function of the COG coordinates \( X_G, Y_G \). In particular, the coefficients \( b_R, b_L \) has the following expression: \( b_R = \frac{s}{2} - Y_G \) and \( b_L = -\frac{s}{2} - Y_G \) whereas the coefficients \( a_i \) can be calculated as follows: \( a_1 = -(x_{a1} - x_{a2})/2 + X_G = c_1 + X_G, \) \( a_2 = -(x_{a2} - x_{a3})/2 + X_G = c_2 + X_G, \) \( a_3 = (x_{a2} - x_{a3})/2 + X_G = c_3 + X_G \) and \( a_4 = (x_{a1} - x_{a4})/2 + X_G = c_4 + X_G \).
\[ X_G = c_4 + X_G. \] Finally, the moment equilibrium equations can be re-written as: \( C \hat{G} = d \) where \( C \in \mathbb{R}^{2 \times 2}, \hat{G}, d \in \mathbb{R}^2 \) and

\[
C = \begin{bmatrix} \sum_{i=1}^{n_{tot}} (\hat{N}_{Ri} + \hat{N}_{Li}) & 0 \\ -\sum_{i=1}^{n_{tot}} (\hat{N}_{Ri} + \hat{N}_{Li}) & 0 \end{bmatrix}
\]

\[
\hat{G} = \begin{bmatrix} \hat{X}_G \\ \hat{Y}_G \end{bmatrix} \quad d = \begin{bmatrix} 0 \\ \sum_{i=1}^{n_{tot}} (\hat{N}_{Li} - \hat{N}_{Ri}) \end{bmatrix}
\]

The values of the estimated longitudinal \( \hat{X}_G \) and lateral \( \hat{Y}_G \) coordinates of the center of gravity \( G \) can be computed by simply inverting the \( C \) matrix: \( \hat{G} = C^{-1}d \).

## 6 PERFORMANCE AND ROBUSTNESS OF THE WIM ALGORITHM

This chapter describes the performance of the WIM algorithm in estimating the vertical axle or wheel loads \( \hat{N}_{Ri}, \hat{N}_{Li} \), and the coordinates of the vehicle center of gravity \( X_G \) and \( Y_G \), starting from the knowledge of the vertical loads on the sleepers \( F_z \). The WIM algorithm has been tested in different simulation campaigns. The first one considers variable vehicle parameters (mass and velocity), the second one tests the algorithm robustness against track parameters uncertainties.

### 6.1 Estimation of the vertical wheel loads \( \hat{N}_{Ri}, \hat{N}_{Li} \) and center of gravity \( G \) with variable vehicle car body mass \( M \), speed \( V \), cut-off frequency \( f_n \)

The vertical forces on the sleepers \( F_{zrk}^{fn}(t) = F_{zrk}^{fp}(x_{rk}, t) \) and \( F_{izk}^{fn}(t) = F_{izk}^{fp}(x_{izk}, t) \) evaluated through the physical model of the railway track are compared to the vertical forces on the sleepers \( F_{zrkapp}^{fn}(t) = F_{zrkapp}^{fp}(x_{rk}, t) \) and \( F_{izkapp}^{fn}(t) = F_{izkapp}^{fp}(x_{izk}, t) \) estimated by means of the WIM algorithm. To perform the comparison, an extensive simulations campaign has been carried out. In particular the dependence of the estimated relative errors \( e_{ri}^{sim} = \frac{\hat{N}_{ri} - N_{ri}}{N_{ri}} \) and \( e_{li}^{sim} = \frac{\hat{N}_{li} - N_{li}}{N_{li}} \) on the vehicle speed \( V \), car-body mass \( M \) and the cut frequency \( f_n \) of the physical system is investigated.

In Tab. 1 the considered variation ranges for the previous quantities are reported together with the resolutions adopted for the range discretization (\( \Delta V \), \( \Delta M \), \( \Delta f_n \)), where \( N_t, N_M, N_f \) represent respectively the number of simulated values of \( V, M \) and \( f_n \).

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Min.</th>
<th>Max.</th>
<th>( N_{sim} )</th>
<th>( \Delta )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Velocity (m s(^{-1}))</td>
<td>10</td>
<td>40</td>
<td>4</td>
<td>( \Delta V = \frac{(V_{max} - V_{min})}{(N_t - 1)} )</td>
</tr>
<tr>
<td>Car-body Mass (t)</td>
<td>20</td>
<td>50</td>
<td>4</td>
<td>( \Delta M = \frac{(M_{max} - M_{min})}{(N_M - 1)} )</td>
</tr>
<tr>
<td>Frequency (s(^{-1}))</td>
<td>10</td>
<td>40</td>
<td>4</td>
<td>( \Delta f_n = \frac{(f_{max} - f_{min})}{(N_f - 1)} )</td>
</tr>
</tbody>
</table>

In the present research activity, the layout of the adopted measurement station consists in \( N_m = 3 \) measure points on both rail side (\( x_{R1} = x_{L1} = 33 \) m, \( x_{R2} = x_{L2} = 34.2 \) m and \( x_{R3} = x_{L3} = 38.4 \) m). Fig. 5 illustrates both simulated \( F_{zrk}^{fn}(t) \) \( F_{izk}^{fn}(t) \) and approximated \( F_{zrkapp}^{fn}(t) \) \( F_{izkapp}^{fn}(t) \) right and left vertical forces acting on the second measurement point on the right side of the sleeper (\( x_{L2} = 34.2 \) m) relative to a simulation performed at \( V = 40 \) m s\(^{-1}\), with a car-body mass \( M = 50 \) t and with a cut frequency \( f_n = 20 \) s\(^{-1}\). The figure shows a good comparison between the simulated and estimated quantities, confirming the accuracy of the WIM algorithm. The global performance of the WIM algorithm have been studied by considering the maximum relative error \( e_{max}^{sim}(V, M, f_n) \):

\[
e_{max}^{sim} = \max_{t \leq T_{tot}} \left( \max(|e_{ri}^{sim}|, |e_{li}^{sim}|) \right)
\]
Figure 5. Vertical load acting at $x_{r2} = 34.2$; comparison between the value resulting from the physical model $F_{fn}^{zr2}(t)$, $F_{fn}^{zl2}(t)$ and the one obtained according to the quasi-linearity hypothesis $F_{fn}^{zr2\ app}(t)$, $F_{fn}^{zl2\ app}(t)$

and analysing the maximum error values $\epsilon_{\text{max}}(V, M, f_n)$. The following figures (Figs. 6, 7 and 8) show a comparison between the percentage errors, and their behaviour as a function of speed and cut-off frequencies; each graph is related to a different value of vehicle car body mass $M$.

Figure 6. Relative error, function of speed $V$ and cut-off frequency $f_n$, $M=20$ t.

Figure 7. Relative error, function of speed $V$ and cut-off frequency $f_n$, $M=30$ t.

These results show how the estimation of vertical loads becomes more difficult, in front of an increase of the travel speed, and a decrease of the cut frequency $f_n$. The vehicle center of gravity $G_B$ has been varied in both longitudinal and lateral direction, in order to simulate a vehicle unbalanced load. The displacement values satisfy the limits imposed by the real dimensions of the vehicle car body, with the aim of reproducing a real scenario. Firstly the actual longitudinal $X_G$, and then the lateral $Y_G$ position of the center of mass $G$ of the train have been varied by changing the position of the center of mass of the car-body $G_B$ in the physical model of the train, according to the values shown in Fig. 9. The test campaign has been performed considering a car-body mass value $M=10$
Figure 8. Relative error, function of speed \( V \) and cut-off frequency \( f_n \), \( M=40 \) t.

\( t \), a cut frequency \( f_n = 20 \) s\(^{-1} \) and four different values of the vehicle velocity (\( V=10,20,30,40 \) m s\(^{-1} \)).

Figure 9. Variation of the \( G_B \) positions

Table 2. Percentage relative errors on \( \hat{X}_G \) with different speed and longitudinal displacement \( X_G \) of center of gravity

<table>
<thead>
<tr>
<th>Position ( X_G )</th>
<th>Speed ( 10 ) m s(^{-1} )</th>
<th>Speed ( 20 ) m s(^{-1} )</th>
<th>Speed ( 30 ) m s(^{-1} )</th>
<th>Speed ( 40 ) m s(^{-1} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( X_G = 0 )</td>
<td>0.101 %</td>
<td>0.155 %</td>
<td>0.264 %</td>
<td>0.260 %</td>
</tr>
<tr>
<td>( X_G = +1 )</td>
<td>0.194 %</td>
<td>0.159 %</td>
<td>0.199 %</td>
<td>0.423 %</td>
</tr>
<tr>
<td>( X_G = +2 )</td>
<td>0.194 %</td>
<td>0.241 %</td>
<td>0.249 %</td>
<td>0.329 %</td>
</tr>
<tr>
<td>( X_G = +3 )</td>
<td>0.168 %</td>
<td>0.290 %</td>
<td>0.192 %</td>
<td>0.370 %</td>
</tr>
</tbody>
</table>

Tab. 2 indicates the percentage relative errors on the estimation of longitudinal displacement \( X_G \) of center of gravity.

The second simulation campaign is focused on the lateral displacement \( Y_G \) of the vehicle center of gravity always according to the real dimensions of the car body. Table. 3 summarizes the behaviour of the errors as a function of the speed. Tab. 2 and 3 show a limited increasing of the errors with increasing speed, highlighting the accuracy of the algorithm.
Table 3. Percentage relative errors on \( \hat{Y}_G \) with different speed and lateral displacement \( Y_G \) of center of gravity

<table>
<thead>
<tr>
<th>Position ( m )</th>
<th>Speed ( 10 \text{ m s}^{-1} )</th>
<th>Speed ( 20 \text{ m s}^{-1} )</th>
<th>Speed ( 30 \text{ m s}^{-1} )</th>
<th>Speed ( 40 \text{ m s}^{-1} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( Y_G = 0 ) ( m )</td>
<td>0.131 %</td>
<td>0.139 %</td>
<td>0.199 %</td>
<td>0.272 %</td>
</tr>
<tr>
<td>( Y_G = +0.1 ) ( m )</td>
<td>0.151 %</td>
<td>0.152 %</td>
<td>0.251 %</td>
<td>0.278 %</td>
</tr>
<tr>
<td>( Y_G = +0.2 ) ( m )</td>
<td>0.194 %</td>
<td>0.241 %</td>
<td>0.284 %</td>
<td>0.213 %</td>
</tr>
<tr>
<td>( Y_G = +0.3 ) ( m )</td>
<td>0.168 %</td>
<td>0.290 %</td>
<td>0.187 %</td>
<td>0.222 %</td>
</tr>
<tr>
<td>( Y_G = +0.4 ) ( m )</td>
<td>0.168 %</td>
<td>0.290 %</td>
<td>0.187 %</td>
<td>0.222 %</td>
</tr>
</tbody>
</table>

6.2 Analysis of the WIM algorithm robustness

The robustness analysis has been implemented imposing a uncertainty on the stiffness and vertical dampers of sleepers \( (K_{z,sl} \text{ and } C_{z,sl}) \) and ballast \( (K_{z,bal} \text{ and } C_{z,bal}) \), (the nominal values are reported in Tab. 5, and the uncertainties in Tab. 4).

Table 4. Uncertainties on stiffness and damping of sleeper and ballast

<table>
<thead>
<tr>
<th>Simulation</th>
<th>Parameters</th>
<th>Min.</th>
<th>Max.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Run1</td>
<td>( K_{z,sl}, C_{z,sl} )</td>
<td>-10 %</td>
<td>+10%</td>
</tr>
<tr>
<td>Run2</td>
<td>( K_{z,bal}, C_{z,bal} )</td>
<td>-10 %</td>
<td>+10%</td>
</tr>
<tr>
<td>Run3</td>
<td>( K_{z,sl}, C_{z,sl}, K_{z,bal}, C_{z,bal} )</td>
<td>-10 %</td>
<td>+10%</td>
</tr>
</tbody>
</table>

Table 5. Nominal characteristics of vertical stiffness and damping of sleepers and ballast

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Meaurement Unit</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vertical stiffness of sleeper</td>
<td>N m(^{-1})</td>
<td>( 2 \times 10^8 )</td>
</tr>
<tr>
<td>Vertical damping of sleeper</td>
<td>N s m(^{-1})</td>
<td>( 5.6 \times 10^5 )</td>
</tr>
<tr>
<td>Vertical stiffness of ballast</td>
<td>N m(^{-1})</td>
<td>( 1 \times 10^9 )</td>
</tr>
<tr>
<td>Vertical damping of ballast</td>
<td>N s m(^{-1})</td>
<td>( 1 \times 10^6 )</td>
</tr>
</tbody>
</table>

Both of tests have been performed for one value of vehicle mass \( M \) (equal to 30 t) and for two values of speed \( V \) \( (20 \text{ m s}^{-1} \text{ and } 40 \text{ m s}^{-1}) \). Results are reported in Tab. 6 for the sleepers and Tab. 7 for the ballast. These results show again how the architecture of the proposed WIM algorithm proposed is able to make an accurate and reliable estimation of vertical loads on wheels in different operative conditions, even introducing high uncertainty on track parameters.

Table 6. Relative percentage error on the loads estimation with uncertainty on sleepers

<table>
<thead>
<tr>
<th>Architecture 1</th>
<th>Speed ( 20 \text{ m s}^{-1} )</th>
<th>Speed ( 40 \text{ m s}^{-1} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( K_{z,sl} \rightarrow + 10 % \text{ and } C_{z,sl} \rightarrow + 10 % )</td>
<td>4.8 %</td>
<td>5.7 %</td>
</tr>
<tr>
<td>( K_{z,sl} \rightarrow - 10 % \text{ and } C_{z,sl} \rightarrow - 10 % )</td>
<td>5.2 %</td>
<td>5.9 %</td>
</tr>
</tbody>
</table>
Table 7. Relative percentage error on the loads estimation with uncertainty on ballast

<table>
<thead>
<tr>
<th>Architecture 1</th>
<th>Speed 20 m s$^{-1}$</th>
<th>Speed 40 m s$^{-1}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K_{z, bal}$ → + 10% $C_{z, bal}$ → + 10%</td>
<td>2.9%</td>
<td>2.63%</td>
</tr>
<tr>
<td>$K_{z, bal}$ → - 10% $C_{z, bal}$ → - 10%</td>
<td>3.4%</td>
<td>3.4%</td>
</tr>
</tbody>
</table>

7 CONCLUSIONS

In this paper an innovative WIM algorithm has been presented with the aim of estimating the vertical wheel loads $\hat{N}_{Ri}, \hat{N}_{Li}$ and the longitudinal $\hat{X}_G$ and lateral $\hat{Y}_G$ coordinates of the gravity center of railway vehicles. The proposed algorithm has the advantage to manage both real experimental and simulated data (when experimental data are not available). Once verified the good performances of the estimation algorithm with different vehicle parameters (speed, car body mass and center of gravity), its robustness has been tested, against track uncertainty on the stiffness and dampers elements of ballast, sleeper. The obtained results shows a relative percentage error always less than 4 %, highlighting the capability of the presented algorithm to make an accurate and reliable estimation with different operative conditions and uncertainty on track parameters.

REFERENCES


Analysis of Critical Hunting Speed and Running Safety of Conventional Railway Vehicle Truck on Curved Track

Sajeev. R. K* and Sujatha Chandramohan#

Machine Design Section, Department of Mechanical Engineering
Indian Institute of Technology Madras, Chennai 600036, India
r.k.sajeev0891@gmail.com, sujatha@iitm.ac.in

ABSTRACT
Critical hunting speed and running safety are important aspects of the dynamic behaviour of railway vehicles. In this paper, a study of the dynamic behaviour of a single railway vehicle truck running on tangent and curved tracks is carried out using a mathematical model of the combined system. It is seen from literature that there are no reported studies using the combined truck/curved track system in which rail lateral displacement is computed considering two point wheel/rail contact for evaluation of critical hunting speed, as well as derailment due to wheel climb. In this study, the truck consists of the truck frame with suspended wheelsets, while the track is idealized as a laterally flexible rail modelled as a spring mass damper system. The combined truck/track system has 10 degrees of freedom (DOF), consisting of the lateral displacement and yaw angle of the wheelsets and truck frame, as well as the lateral displacement of the front and the rear left and right rails. Equations of motion using a model with single-point and two-point wheel-rail contact are derived. Non-linearities in the wheelset model include the non-linear wheel-rail profile and the friction-creep characteristics of the wheel-rail contact geometry. The longitudinal and lateral primary and secondary suspensions are assumed to have linear stiffness and damping characteristics. A combination of linear Kalker’s theory and non-linear heuristic creep model is adopted to calculate the creep forces. The mathematical equations of motion are solved using fourth order Runge-Kutta method, which requires that the second order differential equations be transformed into a set of first order differential equations. The transformed state space equations are solved in the time domain to obtain the dynamic response of a conventional truck, moving on tracks of various radii. The numerical simulation is done using MATLAB.

Keywords: Critical Hunting Speed, Running Safety, Single/Two Point Contact, Non-linear Creep Model, Curved Track.

1 INTRODUCTION
Critical hunting and running safety are important issues in the dynamic behaviour of railway vehicles. Hunting is a mode of instability, which appears at higher speeds as a self-excited oscillation of combined lateral and yaw swaying motion in the truck when the forward speed attains a certain limit known as critical speed. This speed refers to the minimum speed of the vehicle for a given set of system parameters and suspension characteristics, beyond which hunting appears as an undamped motion of the wheelset, constrained only by the wheel flange and the rail. Suspension elements in a truck change their properties with time and wear due to general deterioration. These changes may lower the critical speed. In order to maintain a safety margin, vehicle suspension design should be based on a critical speed above the intended maximum operating speed.

The running safety of the vehicle is evaluated from the derailment quotient and the offload factor. Derailment quotient is defined as the ratio of the wheel's guiding (lateral) force to the vertical force between the wheels and rails. Offload factor is defined as the ratio of the reduction in vertical force to the static wheel load.

The hunting-instability and running safety problems of railway vehicles on tangent and curved tracks have been intensively investigated in many papers. Cooperrider [1] was the first to note the sources and implications of non-linearities in railway vehicle systems. Nagurka [2], in examining curving behaviour of rail vehicles, is perhaps the first author to incorporate a two-point wheel-rail contact condition and non-linear heuristic model to calculate the creep forces in the equations of motion of the rail vehicle. Ahmadian and Yang [3] investigated the Hopf bifurcation and hunting behaviour in a rail wheelset with single point flange contact and studied the effects of non-linear longitudinal yaw damping on hunting critical speeds. Imtiaz Haque et al. [4] studied a non-linear wheelset model to
predict derailment. Mohan [5] used controllable primary suspensions to improve hunting in railway vehicles moving on tangent tracks; single-point and two-point wheel-rail contact conditions were considered to study the dynamic responses of the rail vehicle components. The sensitivity of the critical speed to various primary suspension stiffness and damping parameters with different wheel conicities was examined. Using a linear creep model, Lee and Cheng [6] derived the governing differential equations of motion of a truck moving on tangent track to study the influences of suspension characteristics and wheel conicity on the hunting speed. Zeng and Wu [7] constructed a model of a high speed railway (HSR) vehicle with 17-DOFs and utilized an efficient numerical method to analyze the critical speed at the Hopf bifurcation point on both straight and curved tracks with various degrees of superelevation. Wu and Zeng [8] studied the influence of flange contact angle, friction coefficient and primary suspension forces on the derailment safety. Cheng and Hsu [9] studied the derailment quotients of a moving railway vehicle with 14-DOF and 27-DOF models. The effect of vehicle speeds on derailment quotients evaluated using both linear and non-linear creep models with various suspension parameters moving on curved tracks have been thoroughly investigated. The effect of vehicle speeds on the derailment quotient for sharp curves and low vehicle speeds are investigated and compared with both linear and non-linear creep models. Lee and Cheng [10] derived the governing differential equation of motion of a truck moving on curved track to study the influences of the vertical and the roll motions of frames on the hunting speed. Lee and Cheng [11] developed a set of non-linear coupled differential equations of motion of 10- and 6-DOF truck system moving on curved tracks to study the influence of the primary suspension parameters on the critical hunting speeds evaluated using linear and non-linear creep models. Hoffmann and Petersen [12] analysed Cooperider’s bogie. The model accommodates the possibility of running the train through a curve and studying the influence of track superelevation.

The basic element of the rail vehicle steering and support system is the truck frame and the wheelset. The contact and friction mechanisms, which develop at the wheel-rail interfaces, have a dominant effect on vehicle curving behaviour. The curving performance of a vehicle is a direct function of the ability of its wheelsets to negotiate a curve. The suspension system plays a major role in determining the curving performance and stability of rail vehicles. Optimized combinations of vehicle suspension have been proposed in literature to control truck frame and wheelset lateral displacement and yaw angles and wheel-rail forces during curve negotiation and thus improve performance. Softening the suspension system reduces the lateral stability of the vehicle on a straight track. To achieve wheelset radial alignment during curve negotiation without degrading dynamic stability, vehicle with optimal suspension has to be used.

From the existing literature, it can be seen that the dynamic response of a single railway vehicle truck moving on curved tracks, with a combination of rail lateral flexibility, heuristic non-linear creep model and single-point and two-point wheel-rail contact conditions, has never been used to study the influence of primary and secondary suspension and effects of wheel conicity, superelevation and curve radius on the critical hunting speed and derailment due to wheel climb; this has been done in the present paper.

In this study, the non-linear coupled differential equations of motion of the railway vehicle truck model are derived using a heuristic non-linear creep model with single-point and two-point wheel-rail contact. In order to investigate the critical hunting speed and running safety, the lateral displacement and yaw angle of each wheelset and truck frame are considered with the flexible tracks. Therefore, a combined 10-DOF model of the vehicle along with track is adopted in this study. For the running safety analysis, the derailment quotients of the front wheelset of the truck, and offload factors of the 10-DOF truck model are calculated using the Runge–Kutta fourth-order method. Lateral creep and normal contact forces are calculated from single/two point creep and normal force equation [2]. In this study it is assumed that the average static axle load of the vehicle is 100kN, and the vertical force on each wheel is taken as 25kN. The effects of vehicle speeds on derailment quotients and offload factors are evaluated using both the linear and non-linear creep models for various radii of tracks. The non-linear critical hunting speed of the vehicle on tracks of different radii is obtained from phase portrait or limit cycle [5]. Finally the effect of primary suspension, wheel conicity, superelevation angle and radius of curvature of the track, on the running safety, is assessed.
2 TRUCK MODELLING

The truck considered for modeling is a conventional truck that consists of two wheelsets connected to a truck frame. The wheelsets are connected to the truck frame by means of primary suspension as shown in Figure 1. The mathematical governing equations of motion of a single truck model developed by Nagurka [2] have been adapted. The combined truck/track system has 10 DOF, consisting of the lateral displacement and yaw angle of the wheelsets and truck frame, as well as the lateral displacements of the front and the rear left and right rails. Figure 2 shows the contact condition at the left wheel as the wheelset displaces laterally. Single-point tread contact (Figure 2a) and single point flange contact (Figure 2c) occur for net excursions less than and greater than the flange clearance respectively. Two-point contact is depicted in Figure 2b, where the rail-head contacts simultaneously both the tread and flange of the left wheel. Wheelset model (Figure 3) includes the non-linear wheel-rail profile and the friction-creep characteristics of the wheel-rail contact geometry. A combination of linear Kalker’s theory and non-linear heuristic creep model is adopted to calculate the creep forces. Both single-point and two-point wheel/rail contact conditions are considered. A simple model of track flexibility is adopted, in which each rail is assumed to have lateral DOF only. The mass-spring-damper model of the rail is shown in Figure 4. In this model, rail rollover or overturning motion and effect of the lateral rail mass, \( m_{rail} \), are neglected. The longitudinal and lateral primary and secondary suspensions are assumed to have linear stiffness and damping characteristics. In Figure 4, \( m_{rail} \), \( y_{rail,L} \) and \( y_{rail,R} \) represent the masses of the rail and lateral displacement of the left and right rails respectively.

\[ y_{w1}, y_{w2} \text{ and } y_t \text{ denote the lateral displacements of the front and rear wheelsets and truck frame respectively; } \Psi_{w1}, \Psi_{w2} \text{ and } \Psi_{wt} \text{ are the yaw motions of the front and rear wheelsets and truck frame respectively.} \]

\[ \begin{align*}
\text{(a) One-Point Tread Contact} & \quad y_w - y_{rail,L} < y_{fc} \\
\text{(b) Two-Point Contact} & \quad y_w - y_{rail,L} = y_{fc} \\
\text{(c) One-Point Flange Contact} & \quad y_w - y_{rail,L} > y_{fc}
\end{align*} \]

Figure 1. Truck model.

Figure 2. Single-point and two-point left wheel-rail contact situations [2].
The two wheelsets are treated as identical, each one having two longitudinal and two lateral springs and dampers. The only coupling between the front and the rear wheelsets is through a rigid truck frame. When the wheelsets move, they produce a resultant lateral force on the truck, which will tend to move the truck in the same direction as the movement of the wheelsets. Also, when a wheelset yaws, the longitudinal spring-damper combination will produce a truck yawing moment, tending to rotate the truck in the same direction. The net force and moment on the truck will be the resultant of the two wheelset forces and moments [13]. Since the present paper is concerned with the response of the truck, the simulations that follow in this study assume that the carbody is fixed. The secondary suspension is typically considerably softer than the primary suspension in order to prevent the truck motions from passing on to the carbody.

Important features of the present model are as described below.

1. The truck frame is connected to the carbody through secondary suspension and the carbody is considered as fixed.
2. The wheelset is assumed to navigate a changing radius right hand curve with a changing superelevation.
3. The track is assumed to be smooth and the wheelset remains in contact with the rail at all times.
4. Kinematic constraint equations due to wheel-rail contact are used to calculate wheelset vertical and roll velocities.
5. Two-point contact is possible on either wheel.
6. The contact point on the wheel translates longitudinally with yaw angle of the wheelset.
7. Each rail is modelled as an effective mass, spring and damper with a lateral DOF.
8. Effects of rail inertia and rail velocity on creepages are neglected.
9. The creep forces are modeled using an approximation, i.e., Kalker’s non-linear creep theory [14] based on Vermeulen and Johnson’s work [15]. This approach known as the modified Vermeulen-Johnson theory includes the effect of spin creep.
10. The normal force equations contain the influence of roll and vertical accelerations.
11. The equations of motion retain non-linear kinematic quantities and gyroscopic terms.
12. The wheelset maintains continuous wheel-rail contact as it transverses a smooth, laterally flexible, right handed curved track.
13. The forward speed of the wheelset, the track curvature and superelevation (or bank) angles are known functions of the distance along the track.

In the present study, For the simplification of the problem, roll angle of the system is written in terms of lateral displacement i.e., \( \phi_w = \frac{\lambda}{a} y_w \) and the wheelset spin is taken as \( \dot{\theta} = \frac{V}{r_0} \) [10]. The analysis considers the important case of two-point wheel-rail contact, which occurs with many common wheel profiles during curving.

3 RUNNING SAFETY ANALYSIS

Safety is one factor which is very crucial in all areas. When it comes to high speed railway vehicles, it is important to study and to investigate the parameters which influence derailment. In this study some of the parameters considered are curve radius, conicity and superelevation of the track. There are two factors that should be considered in evaluating the running safety of railway vehicles. First, the
derailment quotient is defined as the ratio of contact force acting in the lateral direction to that in the vertical direction. Contact force on the left wheel is shown in Figure 5.

![Figure 5. Contact force on left wheel.](image)

The derailment quotient of the left wheel of the front wheelset of the truck is given as

\[
Q_L = \frac{F_{CLY} + F_{NLY}}{F_{CLZ} + F_{NLZ}}
\]

where \(Q_L\) and \(P_L\) are the contact forces acting in the lateral and vertical direction respectively, on the left wheel. \(F_{CLY}\) and \(F_{CLZ}\) are the creep forces in the lateral and vertical directions, respectively; \(F_{NLY}\) and \(F_{NLZ}\) are the normal forces in the lateral and vertical directions, respectively. The other factor is the offload factor defined as the ratio of the reduction in vertical force to the static wheel load given as

\[
\Delta P = \frac{P_S - P_L}{P_S}
\]

where \(\Delta P\) is the reduction in wheel load and \(P_S\) is the static wheel load on the left wheel.

4 NUMERICAL SIMULATION AND PARAMETRIC STUDY

The dynamic time-domain solution of a single railway vehicle truck (Figure 1) moving on a flexible curved track was obtained using MATLAB simulation. It represents a general method of wheelset analysis of both single-point and two-point wheel-rail contact conditions. The simulations were carried out choosing the forward speed of the wheelset as the bifurcation parameter. The critical forward speed was obtained by increasing the speed gradually until the response of the wheelset became marginally stable. Sensitivity of the critical velocity to suspension parameters was studied.

In a truck dynamic analysis, the lateral dynamics are very important since they determine whether or not flanging occurs. The lateral dynamics are essentially decoupled from the vertical and the longitudinal dynamics. Hence, this simulation neglects the vertical and the longitudinal dynamics of the truck frame and wheelset. This assumption eliminates two degrees of freedom for each wheelset and two degrees of freedom for the truck frame, greatly reducing computation time.

It is also assumed that the effective lateral mass of the rail, \(m_{RAIL}\), is zero. This is justified since the rail lateral stiffness and viscous damping forces dominate. Further, it is assumed that the influence of the lateral rail velocity on lateral creepage is negligible. According to Clark et al. [18], this assumption is reasonable since the lateral creep force is generally saturated during flange contact.

In order to study the response of the truck, the carbody was assumed to be fixed. Since the truck modeled for the simulation is a conventional truck, the front and the rear wheelsets have independent degrees of freedom. Hence, the initial conditions of the two wheelsets can have a wide range of combinations [5]. The two wheelsets can be initially at the same position at one of the rails, or they can be at the opposite rails, or anywhere in between (Figure 3). In the present study, both the wheelsets were placed in the same position i.e., at the centre of the track.

The creep coefficients were taken to be the same for both the tread and the flange contact patches. Initial conditions were assumed as zero for both wheelsets and the truck frame. The initial lateral
displacement and velocity of the left and the right rails were also assumed to be zero. The parametric values used for simulation are shown in Table 1.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Description</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>( f_{11} )</td>
<td>Lateral creep force coefficient (N)</td>
<td>( 9.43 \times 10^6 )</td>
</tr>
<tr>
<td>( f_{22} )</td>
<td>Spin creep force coefficient (N.m)</td>
<td>( 1.20 \times 10^5 )</td>
</tr>
<tr>
<td>( f_{12} )</td>
<td>Lateral spin creep force coefficient (N.m^2)</td>
<td>( 1.0 \times 10^3 )</td>
</tr>
<tr>
<td>( f_{13} )</td>
<td>Longitudinal creep force coefficient (N)</td>
<td>( 10.23 \times 10^6 )</td>
</tr>
<tr>
<td>( I_{WY} )</td>
<td>Pitch mass moment of inertia of wheelset (kg.m^2)</td>
<td>130</td>
</tr>
<tr>
<td>( I_{WZ} )</td>
<td>Yaw mass moment of inertia of wheelset (kg.m^2)</td>
<td>761</td>
</tr>
<tr>
<td>( K_{RAIL} )</td>
<td>Effective rail stiffness (N/m)</td>
<td>( 14.6 \times 10^7 )</td>
</tr>
<tr>
<td>( C_{RAIL} )</td>
<td>Effective rail damping (N.s/m)</td>
<td>( 14.6 \times 10^4 )</td>
</tr>
<tr>
<td>( m_w )</td>
<td>Mass of wheelset (kg)</td>
<td>1751</td>
</tr>
<tr>
<td>( a )</td>
<td>Half of track gauge (m)</td>
<td>0.716</td>
</tr>
<tr>
<td>( d_p )</td>
<td>Half-distance between primary longitudinal springs (m)</td>
<td>0.61</td>
</tr>
<tr>
<td>( \lambda )</td>
<td>Wheel conicity</td>
<td>0.125</td>
</tr>
<tr>
<td>( r_0 )</td>
<td>Centred wheel rolling radius (m)</td>
<td>0.3556</td>
</tr>
<tr>
<td>( \mu )</td>
<td>Wheel-rail friction coefficient</td>
<td>0.15</td>
</tr>
<tr>
<td>( y_{FC} )</td>
<td>Wheel-rail flange clearance (m)</td>
<td>0.008</td>
</tr>
<tr>
<td>( R )</td>
<td>Radius of curvature (m)</td>
<td>1000</td>
</tr>
<tr>
<td>( N )</td>
<td>External load on the wheelset at CG (N)</td>
<td>100000</td>
</tr>
<tr>
<td>( \phi_{se} )</td>
<td>Superelevation angle of the rail (rad)</td>
<td>0.0873</td>
</tr>
</tbody>
</table>

### Variable Parameters

| \( C_{PX} \) | Primary longitudinal damping (N.s/m) | 1810 - 41880 |
| \( C_{PY} \) | Primary lateral damping (N.s/m) | 1810 - 45240 |
| \( K_{PX} \) | Primary longitudinal stiffness (N/m) | \( 2.85 \times 10^4 - 2.85 \times 10^6 \) |
| \( K_{PY} \) | Primary lateral stiffness (N/m) | \( 5.85 \times 10^4 - 1.84 \times 10^6 \) |

### 5 RESULTS AND DISCUSSION

The critical velocity of a single truck with the wheels was determined for various values of longitudinal and lateral stiffness of the primary spring by varying the vehicle forward velocity. All initial conditions of the wheelset, truck frame and rail were assumed as zero. The time step for solving the dynamic wheelset equations was automatically chosen by MATLAB. The following time-domain solutions were obtained through simulation for different values of primary stiffness and damping:

1. The lateral displacement and yaw angle of the wheelset and truck frame.
2. The limit cycle for lateral displacement and yaw angle of the wheelset and truck frame.
3. The lateral displacement of the left and the right rails.

Figures 6, 7 and 8 illustrate the lateral displacement and yaw angle of the front wheelset, rear wheelset and truck frame for truck speed below, at and above the critical hunting speed, i.e., at 12, 15.5 and 18 m/s respectively. In this study the flange clearance was considered as 0.008 m and lateral flexibility of wheel flange was considered as 0.001 m; these values helped in computational flexibility. The results indicate that when the truck speed is less than the critical hunting speed, the truck system is stable and the dynamic responses of the wheelset and truck frame are asymptotically stable. At critical speed lateral displacement oscillates regularly. Above critical speed the oscillation grows and becomes unstable and the lateral displacement oscillates irregularly and derailment takes place.
Figure 6: (a) Lateral displacement and (b) Yaw angle with respect to time below critical speed; ie., \( V = 12 \text{ m/s} \) and \( R = 6250 \text{m} \).

Figure 7: (a) Lateral displacement and (b) Yaw angle with respect to time at critical speed; ie. \( V_c = 15.5 \text{ m/s} \) and \( R = 6250 \text{m} \).

Figure 8: (a) Lateral displacement and (b) Yaw angle with respect to time above critical speed; ie. \( V = 18 \text{ m/s} \) and \( R = 6250 \text{m} \).

Figure 9 shows that on a short radius curve (1000m), the amplitude of lateral displacement of the truck is stabilised/controlled by the wheel/rail two point contact (flanging) as in Figure 2b, compared to the single point contact both only in tread (Figure 2a) or in flange (Figure 2c). The wheel tries to climb the rail till the ratio of lateral force to the vertical force reaches the critical point. The maximum lateral to vertical force ratio at the flanging wheel is given by Nadal’s limit of wheel climb critria [13]. For the
present study, \( L/V = 1.64 \); for derailment of the truck, lateral force should be more than 82 kN. As the flange angle decreases, the lateral force required for the wheelset to climb the rail decreases, increasing the possibility of derailment due to wheel climb. Figure 10 shows a zoomed version of flanging.

Figure 9: Lateral responses of (a) Front wheelset, (b) Rear wheelset, (c) Truck frame and (d) Rails for velocity above critical velocity (\( V = 18 \) m/s, wheel conicity = 0.05) for short radius of curve (\( R = 1000 \)m).

![Figure 9](image1)

Figure 10: Zoomed view of flanging (Two point contact) for (a) Front wheelset and (b) Rear wheelset.

![Figure 10](image2)

Figure 11 is the phase plot of the lateral displacement and yaw angle of the front wheelset. This is for the bogie running above the critical velocity. The figure should be read in a clockwise fashion, and this gives a good indication of how the truck moves on the track. The sensitivity of the truck critical velocity to longitudinal and lateral spring stiffness of primary suspension is depicted in Figures 12 and 13. For the range of values of longitudinal and lateral stiffness considered, higher longitudinal and lateral stiffness yielded considerable increase (improvement) in the truck critical velocity.

Figures 14 and 15 show the sensitivity of truck critical velocity to damping of primary longitudinal and lateral suspensions. For a range of damping values and for fixed lateral and longitudinal spring
constants considered, there is not much effect of damping on the critical velocity of the truck. At very high stiffness, a considerable increase of critical velocity is observed.

Figure 11. State space plot: wheelset yaw (rad) Vs. lateral displacement (m) (V = 18 m/s, conicity = 0.05 and R = 1000m).

Figure 12. Critical velocity vs. longitudinal stiffness of primary suspension.

Figure 13. Critical velocity vs. lateral stiffness of primary suspension.

Figure 14. Critical velocity vs. longitudinal damping of primary suspension.

Figure 15. Critical velocity vs. lateral damping of primary suspension.
Figure 16 shows the maximum amplitude of lateral displacement of front wheelset, rear wheelset and truck frame for different curve radii for superelevation of $2.5^\circ$ and $5^\circ$. It is seen that for increase in curve radius beyond 3000m, the critical hunting speed for the set of parameters remains unchanged. From this graph it is clear that the critical hunting speed remains the same, but the amplitude decreases for various radii of the curve.

Figure 16 : Lateral displacement of the bogie as a function of curve radius at critical hunting speed $(V_c = 15.5\text{m/s})$ for superelevation (a) $5^\circ$ (a) and (b) $2.5^\circ$.

Figure 17 shows the maximum amplitude of yaw angle of front wheelset, rear wheelset and truck frame for different curve radii for superelevation of $2.5^\circ$ and $5^\circ$. This study shows that for the increase in curve radius beyond 3000m, the critical hunting speed for the set of parameters assumed remains unchanged. From this graph it is clear that the critical hunting speed remains the same, but the amplitude decreases for various radii of curve.

Figure 17: Lateral displacement of the bogie as a function of curve radius at critical hunting speed $(V_c = 15.5\text{m/s})$ and for superelevation (a) $5^\circ$ and (b) $2.5^\circ$.

Figure 18 shows the effect of vehicle speed on derailment quotient and offload factor of railway vehicle running on curved track of radius 6250m, evaluated using linear and non-linear creep models. The result shows that below critical velocity of the vehicle, the safety factor almost remains constant. As the velocity approaches the critical velocity, the vehicle derailment quotient increases and at critical
velocity it is maximum. For further increase in velocity, safety factor remains constant in the case of non-linear analysis. This is due to the occurrence of wheel flanging which stabilizes the system. As the vertical load varies, the risk of derailment increases. The difference in the factors evaluated using the two creep models can be neglected for vehicle speeds lower than the critical velocity. The linear creep model predicts safety factor greater than that predicted by the non-linear model. But in the case of offload factor, at speeds greater than the critical speed, linear analysis shows a lower value than non-linear analysis because the calculation involves creep force and normal force in the vertical direction and the static wheel load, where there is no requirement of saturation constant.

Figure 18: Effect of vehicle speeds on (a) Safety factors including derailment quotients and (b) Offload factors of railway vehicle evaluated using linear and non-linear creep model \((R = 6250m)\)

6 CONCLUSIONS

The mathematical model of a single railway vehicle truck moving with constant speed on a smooth curved track is used to numerically investigate the non-linear dynamics of the truck. Non-linearities in the truck model include the non-linear wheel-rail profile and the friction-creep characteristics of the wheel-rail contact geometry. The model has lateral and yaw DOF for truck frame and for each wheelset and lateral DOF for each rail. The model has the capability to represent single-point/two-point contact, as well as transitions from one to the other. Influence of the primary and secondary suspension (stiffness and damping parameters), radius of curvature and wheel conicity on critical hunting velocity is examined.

The results indicate the following points.

The critical velocity (from non-linear analysis) of the bogie has to be placed higher than the running range when negotiating a curve.

On a short radius curve, the amplitude of lateral displacement and yaw angle are seen to be stabilised/controlled by the wheel-rail two point contact compared to the case with single point contact, in tread only or only in flange.

It is observed that lateral displacement and yaw angle are lower for higher radii of curve; below critical speed the amplitude dies out and at critical speed the amplitude remains constant. Above the critical velocity, the wheel climb occurs. In short radius curve, sudden periods of robustness occur as the wheel and rail have a continuous two point contact that makes the system stable. This may be attributable to centripetal forces stabilizing the system.

It is important to include the two-point contact scenario i.e., \(y_w - y_{rail,L} = y_{fc}\) where the normal forces and frictional creep forces are developed in two points (tread and flange). This helps in predicting more accurate solutions.

The critical velocity of the bogie obtained from non-linear analysis is less than the value obtained from linear analysis. Hence the bogie begins to hunt at lower velocities than the critical velocity obtained from linear analysis. So non-linear critical velocity is often referred
to as the critical velocity. Therefore, non-linear critical speed is preferred in bogie design to avoid potentially dangerous operating conditions.

7 REFERENCES

Bifurcation analysis and hunting behavior of high-speed railway vehicle in a curve

M.Azarinezhad*, M.R.Ghazavi#, S.Rahmanian†

* Mechanical Engineering Department
Tarbiat Modarres University
Tehran, Iran
m.azarinezhad@modares.ac.ir

# Mechanical Engineering Department
Tarbiat Modarres University
Tehran, Iran
ghazavim@modares.ac.ir

† Mechanical Engineering Department
Tarbiat Modarres University
Tehran, Iran
s.rahmanian@modares.ac,ir

ABSTRACT

A vehicle body, two bogie frames and two wheelsets for each bogie frame, are used to describe the railway vehicle model. The system dynamic is modeled by using a 38-DOF system including the longitudinal, lateral, vertical, roll, pitch and yaw displacements. A heuristic nonlinear creep model and an elastic rail model are used to simulate the wheel-rail contact. Stable movement and the corresponding limit cycle motion are investigated. In order to study stability, bifurcation analyses are performed. The effects of the longitudinal displacement and the nonlinear elastic rail model on the lateral stability are investigated. Also, whenever possible, in order to demonstrate the accuracy of results, it compared to the available other author’s results. It is revealed that the longitudinal displacement does not affect on the lateral stability.

Keywords: railway vehicle dynamics, heuristic nonlinear creep model, critical hunting speed, numerical simulation, bifurcation analysis.

1 INTRODUCTION

By coming high speed railway vehicles the lateral displacement of wheelset, can cause large domain vibrations that define vehicle lateral instability. This phenomenon is called hunting. In order to prevent unpleasant incidents, preservation high-speed railway vehicle stability has vital importance. Qualitative change in the nature of the solution when a parameter passes through a critical point hat is defined bifurcation. The bifurcation analysis is a common way to describe stability.

A lot of studies are performed on the stability and bifurcation analysis of railway vehicle. Ahmed and Sankar investigated the effects of elasto-damper coupled wheelset parameters on the linearized stability railway vehicle [1, 2]. Based on the symbolic dynamic, Knudsen et al. [3, 4] performed bifurcation analysis in order to study the effects of the flange and suspension stiffness on the railway vehicle stability. One of the other important researches was carried out by Lee and Cheng [5-9]. They investigated the effect of suspension stiffness on the critical hunting speed in tangent and curved tracks. Their analysis is carried out a range of degrees of freedom system. Zeng and Wu [10], studied the effect of superelevation and curvature on the critical hunting speed with consideration 17-DOF and numerical methods. Zboinski and Dusza [11], studied the wheel–rail profiles, suspension parameters and wear of the wheel–rail profiles on the stability. They [12], investigated the influence of mean rolling radius determination and rail inclinations on the stability. In order to specifying the railway vehicle stability in a curve, the basic procedure presented by their later work [13]. Zhai and Wang [14], presented a direct numerical method to determine the nonlinear hunting speed of railway vehicles. They investigated the difference between the rigid and elastic track model on the critical hunting speed. Based on a piecewise linear function and the Vermeulen-Johnson creep force laws, Gao et al. [15], determined the saddle node bifurcation point and the hopf bifurcation point for four-
axle railway passenger car with consideration 17-DOF. In order to study the influence of nonlinearity on the system stability, Kim and Seok [16], have drawn the bifurcation diagram by multiple scales method. Kim et al. [17], performed stability analysis using Lyapunov’s indirect method of a railway vehicle with consideration 31-DOF. They studied the effect of suspension stiffness and damping coefficients on the critical hunting speed.

The previous authors have not considered the longitudinal displacement and the nonlinear elastic rail model in their researches. In this research, based on the Kalker’s linear theory and the saturation constant, a heuristic nonlinear creep model is defined. The railway vehicle model includes a vehicle body, two bogie frames, and two wheelsets for each bogie frame. With consideration all possible motion (longitudinal, lateral, vertical, rolling, pitching and yawing) 38-DOF is used to describe the system dynamic. In order to solve the coupled and nonlinear equations of motion, Runge-Kutta method of order four is used. Time history, phase portraits and orbital representation are drawn to investigate the stable movement and corresponding limit cycle motion. In order to study stability, bifurcation analyses are performed. In bifurcation analysis, speed is considered as the bifurcation parameter. Influences of the longitudinal displacement and the nonlinear elastic rail on the lateral stability are investigated. Finally, frequency analysis is carried out for the lateral displacement of the leading wheelset in front bogie.

**Nomenclatures**

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>Half of the track gauge</td>
</tr>
<tr>
<td>b_{i1}, b_{i3}</td>
<td>Half of the secondary longitudinal and vertical spring arms</td>
</tr>
<tr>
<td>b_{i2}, b_{i4}</td>
<td>Half of the secondary longitudinal and vertical damper arms</td>
</tr>
<tr>
<td>b_{i1}, b_{i3}</td>
<td>Half of the primary longitudinal and vertical spring arms</td>
</tr>
<tr>
<td>b_{i2}, b_{i4}</td>
<td>Half of the primary longitudinal and vertical damper arms</td>
</tr>
<tr>
<td>C_{p1}, C_{p3}, C_{p2}</td>
<td>Primary longitudinal, lateral and vertical damping coefficients, respectively</td>
</tr>
<tr>
<td>C_{s1}, C_{s3}, C_{s2}</td>
<td>Secondary longitudinal, lateral and vertical damping coefficients, respectively</td>
</tr>
<tr>
<td>f_{i1}, f_{i2}, f_{i3}</td>
<td>Lateral, lateral/spin, spin and longitudinal creep coefficients, respectively</td>
</tr>
<tr>
<td>F_{Li}, F_{Ly}, F_{Lz}</td>
<td>Linear creep force of the left wheel in the longitudinal, lateral and vertical directions, respectively</td>
</tr>
<tr>
<td>F_{Li}^<em>, F_{Ly}^</em>, F_{Lz}^*</td>
<td>Linear creep force of the left wheel given by Kalker’s linear theory in the longitudinal and lateral directions, respectively</td>
</tr>
<tr>
<td>F_{Ri}, F_{Ry}, F_{Rz}</td>
<td>Linear creep force of the right wheel in the longitudinal, lateral and vertical directions, respectively</td>
</tr>
<tr>
<td>F_{Ri}^<em>, F_{Ry}^</em>, F_{Rz}^*</td>
<td>Linear creep force of the right wheel given by Kalker’s linear theory in the longitudinal and lateral directions, respectively</td>
</tr>
<tr>
<td>F_{swi}, F_{sxi}, F_{szi}</td>
<td>Longitudinal suspension forces of the car body, bogies and wheelsets, respectively</td>
</tr>
<tr>
<td>F_{swi}, F_{sxi}, F_{szi}</td>
<td>Lateral suspension forces of the car body, bogies and wheelsets, respectively</td>
</tr>
<tr>
<td>F_{swi}, F_{sxi}, F_{szi}</td>
<td>Vertical suspension forces of the car body, bogies and wheelsets, respectively</td>
</tr>
<tr>
<td>F_{ij}</td>
<td>Flange contact force</td>
</tr>
<tr>
<td>h</td>
<td>Height of the vehicle body mass center above the wheelset mass center</td>
</tr>
<tr>
<td>h_{0}</td>
<td>Height of the secondary suspension above the bogie frame mass center</td>
</tr>
<tr>
<td>h_{G}</td>
<td>Height of the bogie mass center above the wheelset mass center</td>
</tr>
<tr>
<td>I_{cx}, I_{cy}, I_{cz}</td>
<td>Roll, pitch and yaw moments of inertia of the vehicle body, respectively</td>
</tr>
</tbody>
</table>
Roll, pitch and yaw moments of inertia of the bogie frame, respectively
Roll, pitch and yaw moments of inertia of the wheelset, respectively
Primary longitudinal, lateral and vertical stiffnesses, respectively
Vertical and lateral rail stiffnesses, respectively
Secondary longitudinal, lateral, and vertical stiffnesses, respectively
Distance between the vehicle body and the bogie frame mass center
Half of the primary lateral spring and damper arms
Vehicle body, bogie frame and wheelset masses, respectively
Degrees of nonlinearity elastic rail
Linear creep moment of the left wheel in the longitudinal and vertical directions, respectively
Linear creep moment of the left wheel in the vertical direction given by Kalker’s linear theory
Linear creep moment of the right wheel in the longitudinal and vertical directions, respectively
Linear creep moment of the right wheel in the vertical direction given by Kalker’s linear theory
Suspension moments of the car body, bogies and wheelsets, respectively, in the longitudinal direction
Lateral suspension moments of the car body, bogies and wheelsets, respectively
Vertical suspension moments of the car body, bogies and wheelsets, respectively
Normal forces on the left wheel in the longitudinal, lateral and vertical directions, respectively
Normal forces on the right wheel in the longitudinal, lateral and vertical directions, respectively
Nominal wheelset rolling radius
Left-wheel and right-wheel rolling radii, respectively
Contact position vector components on the left wheel, respectively
Contact position vector components on the right wheel, respectively
Radius of the curved track
Forward speed of the railway vehicle
Longitudinal, lateral, vertical displacements, roll, pitch and yaw angles of the car body, respectively
Longitudinal, lateral, vertical displacements, roll, pitch and yaw angles of the bogies, respectively
Longitudinal, lateral, vertical displacements, roll and yaw angles of the wheelsets, respectively
Saturation constant in the heuristic nonlinear creep model
Nonlinearity in the heuristic nonlinear creep model
Nonlinearity of the left wheel in the heuristic nonlinear creep model
Nonlinearity of the right wheel in the heuristic nonlinear creep model
Flange clearance
Contact angles of the left and right wheels, respectively
Wheel conicity
Coefficient of friction
Cant angle

2 VEHICLE SYSTEM

2.1 Railway vehicle model

The vehicle system has been illustrated in Fig. 1. This system includes a vehicle body, two bogie frames and two wheelsets for each bogie frame. Subscripts $p$ and $s$ in this Fig. indicate the primary and secondary suspension. Three spring-damper systems are used to describe each suspension. The primary suspension connected the wheelsets and bogie frames and the secondary suspension connected the vehicle body and bogie frames. 6-DOF was defined for car body and 6-DOF for each bogie frame. By considering constant rotation speed, one can neglect of the pitch motion of the wheelset around $y$-axis, so 5-DOF was defined for each wheelset (Fig. 2). In Fig. 2 the subscripts $i$ and $j$ indicate the bogies location ($i=1$ for front bogie and $i=2$ for rear bogie) and the wheelsets location ($j=1$ for leading wheelset and $j=2$ for trailing wheelset). Coordinate axes are placed at the mass center of the car body, bogie and wheelset. So that the $x$, $y$ and $z$ axes are in the longitudinal, lateral and vertical directions.

Fig. 1. Vehicle system [16].

Fig. 2. Degrees of freedom system
2.2 Wheel-rail contact model

In this paper, the rail is considered elastic. For this purpose, two springs with high stiffness are used to describe the rail (Fig. 3).

![Dynamic model for the contact structure between the rail and wheel](image)

Fig. 3. Dynamic model for the contact structure between the rail and wheel [16].

By combining the Kalker’s linear theory and the saturation constant, a heuristic nonlinear creep model is defined. So, the creep forces and moments in the heuristic nonlinear creep model are represented as [18]

$$
F_{p_{ij}}^n = \alpha_i F_{p_{ij}}^n \\
F_{p_{ij}}^n = \alpha_i F_{p_{ij}}^n \\
M_{p_{ij}}^n = \alpha_i M_{p_{ij}}^n
$$

(1)

Where, the subscript $P$ indicate the wheels location ($P=L$ for left wheel and $P=R$ for right wheel).

By the Kalker’s linear theory, the creep forces and moments are represented as [19]

$$
F_{L_{ij}}^* = -\frac{f_{33}}{V} \left( V \left( 1 + \frac{a}{R_y} - \frac{r_L}{r_0} \right) - a\psi_{w_{ij}} \right) \\
F_{L_{ij}}^* = -\frac{f_{11}}{V} \left( \gamma_{wij} + r_L\phi_{wij} - V\psi_{wij} \right) - \frac{f_{12}}{V} \left( \psi_{wij} - \frac{V}{R_y} + \frac{V}{r_0} \delta_L \right) \\
M_{L_{ij}}^* = \frac{f_{12}}{V} \left( \gamma_{wij} + r_L\phi_{wij} - V\psi_{wij} \right) - \frac{f_{22}}{V} \left( \psi_{wij} - \frac{V}{R_y} + \frac{V}{r_0} \delta_L \right) \\
F_{R_{ij}}^* = -\frac{f_{33}}{V} \left( V \left( 1 - \frac{a}{R_y} - \frac{r_R}{r_0} \right) + a\psi_{wij} \right) \\
F_{R_{ij}}^* = -\frac{f_{11}}{V} \left( \gamma_{wij} + r_R\phi_{wij} - V\psi_{wij} \right) - \frac{f_{12}}{V} \left( \psi_{wij} - \frac{V}{R_y} + \frac{V}{r_0} \delta_R \right) \\
M_{R_{ij}}^* = \frac{f_{12}}{V} \left( \gamma_{wij} + r_R\phi_{wij} - V\psi_{wij} \right) - \frac{f_{22}}{V} \left( \psi_{wij} - \frac{V}{R_y} + \frac{V}{r_0} \delta_R \right)
$$

(2)

By transferring the Kalker’s creep forces and moments, the linear creep forces and moments are represented as [19]

$$
F_{L_{ij}} = F_{L_{ij}}^* - F_{L_{ij}}^*\psi_{wij} \\
F_{R_{ij}} = F_{R_{ij}}^* - F_{R_{ij}}^*\psi_{wij}
$$
\[ F_{Lxij} = F_{Lxij}^* \psi_{wij} + F_{Lxij}^* \]
\[ F_{Rxij} = F_{Rxij}^* \psi_{wij} + F_{Rxij}^* \]
\[ F_{Lzij} = F_{Lzij}^* (\delta_L + \varphi_{wij}) \]
\[ F_{Rzij} = -F_{Rxij}^* (\delta_R - \varphi_{wij}) \]
\[ M_{Lxij} = M_{Lxij}^* (\delta_L + \varphi_{wij}) \psi_{wij} \]
\[ M_{Rxij} = -M_{Rzij}^* (\delta_R - \varphi_{wij}) \psi_{wij} \]
\[ M_{Lzij} = M_{Rzij}^* \]
\[ M_{Rxij} = M_{Rxij}^* \]

In the nonlinear creep formulation (Eq. (1)), the saturation constant \( \alpha \) is represented as [17]
\[ \alpha_{ij} = \frac{1}{2\beta_y} \left( \beta_y - \frac{1}{3} \beta_y^2 + \frac{1}{27} \beta_y^3 \right) \left( \tanh \left( 10^4 \left( 3 - \beta_y \right) \right) + 1 \right) + \frac{1}{2\beta_y} \left( \tanh \left( 10^4 \left( \beta_y - 3 \right) \right) + 1 \right) \]

Where, \( \beta \) is the nonlinearity constant that represented by [19]
\[ \beta_y = \beta_{Rxij} + \beta_{Lxij} \]
\[ \beta_{pij} = \frac{\sqrt{\left( F_{pij}^* \right)^2 + \left( F_{pij}^* \right)^2}}{\mu \sqrt{\left( N_{pij} \right)^2 + \left( N_{pij} \right)^2}} \]

The normal forces are represented by
\[ N_{Lxij} = -K_{xij} \left( z_{wij} - \lambda y_{wij} + a \varphi_{wij} \right) \]
\[ N_{Rxij} = -K_{xij} \left( z_{wij} + \lambda y_{wij} - a \varphi_{wij} \right) \]
\[ N_{Lzij} = -N_{Lxij} \tan (\delta_L + \varphi_{wij}) \cos \psi_{wij} \]
\[ N_{Rxij} = N_{Rxij} \tan (\delta_R - \varphi_{wij}) \cos \psi_{wij} \]
\[ N_{Lzij} = -N_{Lxij} \tan (\delta_L + \varphi_{wij}) \sin \psi_{wij} \]
\[ N_{Rxij} = N_{Rxij} \tan (\delta_R - \varphi_{wij}) \sin \psi_{wij} \]

The flange contact force can be represented by
\[ F_{wij} = K_{xij} \left( y_{wij} - \delta \right)^n \left( \tanh \left( 10^4 \left( y_{wij} - \delta \right) \right) + 1 \right) + \frac{K_{xij} \left( y_{wij} + \delta \right)^n}{2} \left( \tanh \left( 10^4 \left( y_{wij} + \delta \right) \right) + 1 \right) \]

In Eqs. (6) and (7), \( n \) is the degrees of rail nonlinearity. In this research, \( n=1 \) and \( n=3 \) show the linear and nonlinear elastic rail model.

By the geometric Structure of the vehicle-rail system, the position vectors is represented as [19]
\[ R_{Rxij} = a \psi_{wij} \]
\[ R_{Rxij} = a + r_R \varphi_{wij} \]
\[ R_{Rxij} = -a \varphi_{wij} - r_R \]
\[ R_{Lxij} = -a \psi_{wij} \]
\[ R_{Lxij} = a + r_R \varphi_{wij} \]
Finally, one can assume
\[ \delta_L = \delta_R = \lambda \]
\[ \frac{1}{2}(r_L - r_R) = \lambda y_{wi} \]
\[ \frac{1}{2}(r_L + r_R) = r_0 \]  

3 EQUATIONS OF MOTION

3.1 Car body
The equations of motion for the car body can be represented as follows
\[ m_c \ddot{x}_c = F_{xc} \]
\[ m_c \left( \ddot{y}_c - \frac{V^2}{R_y} \right) = -m_c g \phi_{sc} + F_{yc} \]
\[ m_c \left( \ddot{z}_c + \frac{V^2 \phi_{sc}}{R_y} \right) = -m_c g + F_{zc} \]
\[ I_{cy} \ddot{\phi}_c = M_{yc} \]
\[ I_{cz} \ddot{\gamma}_c = M_{zc} \]
\[ I_{cy} \ddot{\psi}_c = M_{yc} \]
Where, the subscripts c and s indicate the car body and the suspension system [17].

3.2 Bogie frame
The equations of motion for the bogie frame can be represented as follows
\[ m_{ti} \ddot{x}_{ti} = F_{xsti} \]
\[ m_{ti} \left( \ddot{y}_{ti} - \frac{V^2}{R_y} \right) = -m_{ti} g \phi_{st} + F_{ysti} \]
\[ m_{ti} \left( \ddot{z}_{ti} + \frac{V^2 \phi_{st}}{R_y} \right) = -m_{ti} g + F_{zsti} \]
\[ I_{cx} \ddot{\phi}_{ti} = M_{xsti} \]
\[ I_{cy} \ddot{\gamma}_{ti} = M_{ysti} \]
\[ I_{cz} \ddot{\psi}_{ti} = M_{zsti} \]
Where, the subscript t indicates the bogie frame.

3.3 Wheelset
The equations of motion for the wheelset can be represented as follows
\[ m_w \ddot{x}_{wij} = F_{Lwij}^{n} + F_{Rwij}^{n} + N_{Lwij} + N_{Rwij} + F_{xwij}^{n} \]
\[ m_w \left( \ddot{y}_{wij} - \frac{V^2}{R_y} \right) = -m_w g \phi_{se} + F_{Lwij}^{n} + F_{Rwij}^{n} + N_{Lwij} + N_{Rwij} + F_{ywij}^{n} - F_{tij}^{n} \]
\[ m_w \left( \ddot{z}_{wij} + \frac{V^2 \phi_{se}}{R_y} \right) = -m_w g + F_{Lwij} + F_{Rwij} + N_{Lwij} + N_{Rwij} + F_{zwij}^{n} \]
\[ I_{\omega w} \ddot{\psi}_{wij} + \frac{I_{\omega w} V \dot{\varphi}_{wij}}{r_0} = R_{\omega j} F_{\omega wij}^w - R_{\omega j} F_{\omega rij}^w + R_{\omega j} F_{\omega Lij}^w - R_{\omega j} F_{\omega Lyij}^w + R_{\omega j} N_{\omega Lyij}^w + R_{\omega j} N_{\omega Lij}^w + M_{\omega Lij}^w + M_{\omega rij}^w + M_{\omega wij}^w \]

Where, the subscript \( w \) indicates the wheelset.

4 NUMERICAL ASPECTS

The equations of motion (10), (11) and (12) can be represented in the following form:

\[ \dot{x}(t) = F(x(t)) \]  

Where \( x(t) \) is a vector with 76-dimensional and \( F(x(t)) \) denote a nonlinear vector with 76-dimensional. In order to solve the Eq. (13) zero initial condition, the parameters values listed in Appendix 1 and Runge-Kutta method of order four are used.

5 RESULT AND DISCUSSION

5.1 Stable movement and corresponding limit cycle motion

Numerical simulations are performed for 38-DOF system and linear elastic rail model. The lateral displacement of the leading wheelset in the front bogie for velocities of motion that is equal 150(km/h), is represented in Fig. 4(a). Under these conditions (150(km/h) velocities) the vibrations tend to the equilibrium point for freely long time behavior. The lateral displacement phase portrait for the leading wheelset in the front bogie are illustrated in Figs. 4(b), respectively. One can conclude from these Figs. that by starting on the initial state, stable motion reaches for freely long time behavior. By increasing velocity, the vibrations amplitude increase. For speeds less than the critical hunting speed, the vibrations go away after a long time.

In phase space, a closed path that any closed path is not created around it is called limit cycle. Stable limit cycle is important, because they create the systems that that oscillate even without external forces. Limit cycles are created in two-dimensional nonlinear systems. Limit cycles are unique and any times they are created in phase space, chaos doesn’t occur.

When velocities of motion reaches to 293(km/h), the system would not be stable. This speed is known as the critical hunting speed. The lateral displacement of the leading wheelset in the front bogie for the critical hunting speed is illustrated in Fig. 5(a). In this Fig. the vibrations amplitude is a constant value for a long time. The lateral displacement phase portrait for the leading wheelset in the front bogie are shown in Figs. 5(b), respectively. In these Figs. by starting at the initial state, the vibrations tend to the limit cycle for freely long time behavior. In fact, the limit cycle or hunting motions are occurred for first time.

With velocities higher than the critical hunting speed, the limit cycle motion is preserved and the vibrations amplitude grows lightly for freely long time behavior. For this purpose, the motion with velocities of 297(km/h) is illustrated in Figs. 6(a), 6(b). In real operation, the railway vehicle move at the smaller velocity than the critical hunting speed. The movement at the critical speed increase wheel-rail wear, maintenance costs
and passengers discomfort. Finally, this phenomenon can damage to the track and cause derailment.

Fig. 4. Time history (a), phase portrait (b) for velocities of motion that is equal 150(km/h).

Fig. 5. Time history (a), phase portrait (b) for velocities of motion that is equal 293(km/h).

Fig. 6. Time history (a), phase portrait (b) for velocities of motion that is equal 297(km/h).

5.2 Influence of the longitudinal displacement on the lateral stability
Calculations of previous section were done for 38-DOF system (with consideration the longitudinal displacement). In this section all previous section calculations are done for 31-DOF system (without consideration the longitudinal displacement \([16, 17]\)). Results are revealed that the longitudinal displacement does not affect on the lateral stability. So, analysis 31-DOF and 38-DOF systems have same results on the lateral stability.

5.3 Bifurcation

In order to study the motion at the freely long time, simulations are carried out for different velocities. For each velocity, the leading wheelset’s lateral displacement is plotted versus time (as Figs 4(a), 5(a) and 6(a)). From these Figs. the value of maximum wheelset’s lateral displacement is read in the freely long time. Again, all these calculations are carried out for the nonlinear elastic rail model. One can show these results in the bifurcation diagram (Fig. 7). In this Fig. results is compared to Ref. [17]. The dotted line in this diagram, separate the stable movement from the limit cycle behavior. The speed correspond at the dotted line is called the critical hunting speed. The bifurcation diagram reveals that the critical hunting speed evaluated via the linear elastic rail is greater than that derived using the nonlinear model. These speeds are 293(km/h) and 262(km/h) for the linear and nonlinear elastic rail model, respectively. The values of maximum wheelset’s lateral displacement for the linear elastic rail are equal to the nonlinear model at the speeds lower than 262(km/h). These values are higher for the nonlinear elastic rail than the linear model at the speeds higher than 262(km/h). In order to get higher critical speed, the linear elastic rail has advantage over nonlinear model.

![Fig. 7. Bifurcation diagram for the lateral displacement of leading wheelset](image)

6 CONCLUSION

The railway vehicle dynamic is modeled using a 38-DOF. Based on the Kalker’s linear theory and the saturation constant, a heuristic nonlinear creep model is defined. In order to solve the nonlinear and coupled equations of motion Runge-Kutta method of order four are used. Time history, phase portraits and orbital representation are drawn to study stable movement and corresponding limit cycle motion. The long time behavior indicates that the vibrations tend to the equilibrium state for speeds lower than the critical hunting speed. Stable movement continues while speed is lower than critical
hunting speed. When speed reaches to the critical hunting speed, limit cycle motion occur for first time. With velocities higher than the critical speed, the limit cycle motion is preserved and the vibrations amplitude grows lightly.

Influences of the longitudinal displacement and the nonlinear elastic rail model on the lateral stability are investigated. The bifurcation diagram is used to study the lateral stability. Also, whenever possible, in order to demonstrate the accuracy of results, it compared to the available other author’s results. Result show that the longitudinal displacement does not affect on the lateral stability. Also, that critical hunting speed evaluated via the linear elastic rail is greater than that derived using the nonlinear model. Finally, frequency response is obtained for the lateral displacement of the leading wheelset in the front bogie.

Appendix 1.

Table 1 System parameters and their numerical values [9, 16, 17]

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Half of the track gauge (m)</td>
<td>(a = 0.7465)</td>
</tr>
<tr>
<td>Half of the secondary longitudinal and vertical spring arms (m)</td>
<td>(b_{c1}=b_{c3}=1.21)</td>
</tr>
<tr>
<td>Half of the secondary longitudinal and vertical damper arms (m)</td>
<td>(b_{c2}=b_{c4}=1.21)</td>
</tr>
<tr>
<td>Half of the primary longitudinal and vertical spring arms (m)</td>
<td>(b_{t1}=b_{t3}=0.978)</td>
</tr>
<tr>
<td>Half of the primary longitudinal and vertical damper arms (m)</td>
<td>(b_{t2}=b_{t4}=1.21)</td>
</tr>
<tr>
<td>Primary longitudinal, lateran and vertical damping coefficients (KN.s/m)</td>
<td>(C_{px}=C_{py}=12, C_{pz}=15)</td>
</tr>
<tr>
<td>Secondary longitudinal, lateral and vertical damping coefficients (KN.s/m)</td>
<td>(C_{sx}=200, C_{sy}=30, C_{sz}=80)</td>
</tr>
<tr>
<td>Lateral creep, Spin creep and Longitudinal creep coefficients (N)</td>
<td>(f_{11}=10.2 \times 10^6, f_{22}=16, f_{33}=15 \times 10^6)</td>
</tr>
<tr>
<td>Lateral / spin creep coefficient (N.m^2)</td>
<td>(f_{12}=3120)</td>
</tr>
<tr>
<td>Height of the vehicle body mass center above the wheelset mass center (m)</td>
<td>(h=1.4)</td>
</tr>
<tr>
<td>Height of the secondary suspension above the bogie frame mass center (m)</td>
<td>(h_0=0.03)</td>
</tr>
<tr>
<td>Height of the bogie mass center above the wheelset mass center (m)</td>
<td>(h_G=0.44)</td>
</tr>
<tr>
<td>Roll, pitch and yaw moments of inertia of the vehicle body (Kg.m^2)</td>
<td>(I_{cx}=7.506 \times 10^4, I_{cy}=I_{cz}=2.086 \times 10^6)</td>
</tr>
<tr>
<td>Roll, pitch, and yaw moments of inertia of the bogie frame (Kg.m^2)</td>
<td>(I_{tx}=I_{ty}=I_{tz}=2.26 \times 10^3, I_{1y}=I_{2y}=2.71 \times 10^3, I_{1z}=I_{2z}=3.16 \times 10^3)</td>
</tr>
<tr>
<td>Roll, pitch, and yaw moments of inertia of the wheelset (Kg.m^2)</td>
<td>(I_{wx}=915, I_{wy}=140, I_{wz}=915)</td>
</tr>
<tr>
<td>Primary longitudinal, lateral and vertical stiffnesses (KN/m)</td>
<td>(K_{px}=104, K_{py}=5 \times 103, K_{pz}=750)</td>
</tr>
</tbody>
</table>
Vertical and lateral rail stiffnesses (KN/m) \[ K_{rz} = 62 \times 10^6, K_{ry} = 16.17 \times 10^6 \]

Secondary longitudinal, lateral and vertical stiffnesses (KN/m) \[ K_{sx} = K_{sy} = 150, K_{sz} = 400 \]

Half of the primary lateral spring and damper arms (m) \[ L_{t1} = L_{t2} = 1.2 \]

Distance between the vehicle body and the bogie frame mass center (m) \[ L_c = 9 \]

Vehicle body, bogie frame and wheelset masses (kg) \[ m_c = 34000, m_{t1} = m_{t2} = 3000, m_w = 1400 \]

Radius of the curved track (m) \[ R_y = 2000 \]

Flange clearance (m) \[ \delta = 0.00923 \]

Wheel conicity \[ \lambda = 0.05 \]

Friction coefficient \[ \mu = 0.2 \]

Cant angle (rad) \[ \varphi_{sc} = 0.0873 \]

References


CFD assessment of the effect of windblown sand on a high-speed train

C. Paz, E. Suárez, C. Gil, M. Concheiro

School of Industrial Engineering
University of Vigo
Campus Universitario Lagoas-Marcosende, 36310, Spain
[cpaz, suarez, chgil, mconcheiro]@uvigo.es

ABSTRACT
In this paper, an investigation on the effect of windblown sand on the motion of a high-speed train has been carried out by means of computational fluid dynamics (CFD) simulations on the commercial software ANSYS Fluent. For a better understanding of the interaction between sand and train, external routines were implemented, extending the capabilities of the discrete phase model (DPM). The windblown sand has recently become a problem in the railway industry as a result of the increase of tracks crossing desert areas. Such demanding environments are responsible for a decrease in the aerodynamic performance and the wear of materials. The current study has consisted in a series of DPM simulations injected from a vertical plane with different values of the Stokes number and the volume fraction of sand. The forward head has resulted as the most affected region, receiving more impacts as the Stokes number is higher. Some collisions were noticed as well in the roof of the last cars, which reach the train after being displaced by the forward head. A directly proportional dependency between drag force and both variables (Stokes number and volume fraction) was also revealed, being an 11% the maximum increase for the range of particles tested. The results confirm the feasibility of the presented methodology.

Keywords: High-speed train, CFD, aerodynamics, windblown sand, impact.

1 INTRODUCTION
Aerodynamic effects have long been a matter of concern in the design of high-speed trains [1–3]. Issues resulting from the increased velocity, such as slipstreams [4], crosswinds [5], pressure variations [6] or ballast flying [7] have arisen as very important fields of research in the railway industry. These studies have evaluated not only the flow field around the entire train [8], but also the behaviour of the fluid in certain areas, e.g. the wake [9], both through physical experiments [10] and numerical simulations [11]. Furthermore, the progress of the railway technology and the development of countries, such as those belonging the Arabian Peninsula, has led to the construction of tracks in desert areas and a new problem to cope with: the windblown sand. This factor is responsible for overturning accidents [12], the premature wear of materials [13] and the partial loss of aerodynamic performance. Therefore, a deeper understanding of the behaviour of high-speed trains in such demanding environments turns necessary.

The aforementioned phenomena have their origin in the exchange of energy that occurs when the sand grains collide with the train surface. The characteristics of a collision are complex and well-studied [14], [15], and it was determined that they are feasibly modelled through computational methods. Regarding the CFD simulations, the sand can be solved attending whether to the Eulerian or the Lagrangian model. The first strategy, followed by Xiong et al. [16] to study the performance of trains in sandstorms, treats both phases as fluids interacting with each other. On the other hand, in the Lagrangian frame the grains are individually tracked, determining their trajectory through the air, at the expense of a higher computational effort [17].

The purpose of this paper is to estimate through CFD simulations the diminution of the aerodynamic performance and the characteristics of the impacts during the motion of a high-
speed train in a sand-laden environment, considering different Stokes numbers and volume fractions of the particles.

2 METHODOLOGY

2.1 Aerodynamic simulation

Prior to focus on the interaction with the sand, it is necessary to solve properly the air flow around a high-speed train, because the behaviour of the solid phase will be strongly dependent on it. The geometry selected for this study is a full-scale model of an ETR500, with a total length of 200 m. The model is consisted of two power heads and six intermediate cars, including as well simplified bogies and employed on a single-track ballast and rail (STBR) scenario [18] to reproduce better the flow between the train and the track. The computational domain is extended 30H after the train to consider possible relevant effects appearing in the near-wake, and 8H from the nose to the inlet to let the flow develop [11] (see Figure 1), being H the height from the top of rail to the roof of the train.

![Figure 1. Geometrical model and computational domain.](image)

The meshing process was carried out in the software ANSYS Meshing. The “Cutcell” scheme has been selected to create a mainly hexahedral grid, in which as many cells as possible are placed parallel to the inlet flow. To evaluate the effect of grid resolution, two meshes were generated with a surface mesh size of 30 mm and 50 mm, including both of them a fine boundary layer around the train and on the track.

Transient simulations were performed on the commercial software ANSYS Fluent. The speed of the inlet air was set to 83.33 m/s (300 km/h), a regular velocity for high-speed trains. To avoid the use of sliding meshes, the ground and the track were defined as moving walls with the same velocity. This way, the real relative velocity between the train and the surroundings is reproduced. As turbulent model, k-ε Realizable was considered appropriate because of its flexibility and its acceptable accuracy. The duration of the time-step was calculated using a Strouhal number of 0.14 [19] and solving each disturbance during 30 time-steps.

2.2 Particle assessment

As it was previously mentioned, two different strategies are usually applied in the solution of the solid phase: the Eulerian and the Lagrangian model. In this case, the second option was chosen, which is implemented in Fluent through the DPM model. This strategy was considered appropriate because the volume fraction of sand is very low and the computational resources are sufficient to perform the calculation, since a one-way coupling between phases was used. This means that the sand grains are affected by the behaviour of the air flow, but not in the other
way, that is, the solid particles do not disturb the fluid. The force balance of the DPM model is shown as follows in Equation (1):

$$\frac{d\bar{u}_p}{dt} = \frac{C_{D_p} \text{Re}_p (\bar{u} - \bar{u}_p)}{24} + \frac{\bar{g}(\rho_p - \rho)}{\rho_p} \tau_p$$

where $\bar{u}$ is the fluid phase velocity, $\bar{u}_p$ is the particle velocity, $\rho$ is the fluid density, $\rho_p$ is the density of the particle, $\bar{g}$ is the gravity acceleration, $C_{D_p}$ is the drag coefficient of the particle, $\tau_p$ is the relaxation time of the particle and $\text{Re}_p$ is the relative Reynolds number. $\text{Re}_p$ and $\tau_p$ are defined (Equation (2) and (3)) as the following:

$$\text{Re}_p \equiv \frac{\rho d_p |\bar{u}_p - \bar{u}|}{\mu}$$

$$\tau_p = \frac{\rho_p d_p^2}{18 \mu}$$

where $\mu$ is the molecular viscosity of the fluid and $d_p$ is the particle diameter.

The sand grains have been injected in the domain from a vertical plane parallel to the inlet placed in front of the forward head of the train. The dimensions of this plane have been adjusted to the train’s front area to avoid the tracking of useless particles that do not collide the model.

The characterisation of the particles has consisted in the setting of two main parameters: the Stokes number and the volume fraction. On the one hand, according to the International Scale for the identification and classification of soils ISO 14688-1, particle sizes corresponding to the range of medium and fine sand were selected. This led to Stokes numbers ranging from 1 to 50. On the other hand, the volume fraction of sand in air was chosen using the studies done by Lui and Dong [20]. In a still atmosphere, the presence of suspended sand grains with the chosen sizes is almost negligible. However, the passing of the high-speed trains accelerates the surrounding air so much that it is capable to elevate the particles from the floor, creating a sand cloud. In tracks with a high density of trains, this cloud remains in suspension until the passing of a later train, which would suffer the consequences of this harsh environment.

To evaluate the impact of the particles on the train surface, external routines based on the impulse equation were implemented to extend the capabilities of the DPM model. These routines allow for the estimation of the force exerted by the sand in the collision, which increase the aerodynamic drag force that the train should normally overcome, and give a representation through contours of the location of these impacts, determining the regions which suffer a larger wear.

3 RESULTS

3.1 Validation

The results of the aerodynamic simulation, without considering the sand, have been compared with the work published by Rocchi et al. [21], in which several experimental and numerical tests were carried out to determine the velocity profile in the underbody of an ETR500. The comparison between both studies shows a good agreement, especially in the nose region of the forward head, which is very important because this is the region where the majority of the sand grains would collide the train. In addition, it was noticed that the finer mesh offers a better resolution around the bogies, being therefore the selected one for the DPM simulation.
3.2 Distribution of the impacts

The evaluation of the solid phase has consisted in a series of DPM simulations with different values both of the Stokes number of the particles and the volume fraction of the sand. The results show that the most affected region is the forward head, which is collided by the majority of the sand grains. In addition, the number of impacts grows with the Stokes number of the particles, as it can be seen in Figure 2. That happens because the movement of the particles with larger Stokes numbers is less affected by the fluid and, therefore, maintains its rectilinear trajectory instead of being displaced upwards and to the sides, as the smaller particle do. The contours of impact probability show as well that, around the nose, the impacts are distributed along a line in the vertical and lateral direction and, in the flat region between the windshield and the nose, a radial decrease of impacts appear.

![Figure 2. Distribution of impacts in the forward power head for a volume fraction of 4E-5 and Stokes number of 1.96, 17.66 and 49.06, respectively.](image)

Apart from the collisions in the forward head, it was noticed that in the rear part of the train some impacts appear, especially in the roof. The particles that, when reaching the forward head, are displaced upwards, return later to the surface of the train due to the effect of the gravity and the air stream. The images of Figure 3 shows that the impact probability depends strongly on the Stokes number too. For lower values, almost no particle collides the rear part of the train; for medium values, many sand grains reach the train surface and the impacts are mainly located in the last cars; and for high values of Stokes number, the collisions are distributed along the entire train. It must be taken into account that the scale is relative to the maximum value obtained in each case, and has nothing to do with the scale of Figure 1.

![Figure 3. Distribution of impacts in the rear extreme of the train for a volume fraction of 4E-5 and Stokes number of 1.96, 17.66 and 49.06, respectively.](image)

3.3 Measurement of the force
The results of the DPM simulations (see Figure 4) show that the drag resistance caused by the presence of windblown sand depends both on the Stokes number and the volume fraction of particles. Both graphs represent the drag coefficient ratio, defined as the proportion between the drag resistance in a sand-laden environment and the aerodynamic drag in clean air. The curves show an increase of the drag coefficient ratio with the Stokes number and the volume fraction. In addition, this increase seems to follow a logarithmic profile in the first case, and a linear one in the second case, where the slope of the lines is more pronounced for higher Stokes number. For the range of particles tested, the maximum variation in drag corresponds to about an 11% of the aerodynamic value, although most of the cases do not exceed a 2%. This increase cannot be neglected in any case, because in lines where the windblown sand is a common phenomenon, this would imply a greater fuel consumption and, therefore, higher operational costs.

**Figure 4.** Graphs of the drag coefficient ratio as a function of the Stokes number and the volume fraction of the particles.

### 4 CONCLUSIONS

In this paper, CFD simulations were carried out to evaluate the effect of windblown sand in the motion of a high-speed train. A transient simulation of the continuous phase was performed and validated against experimental data, being afterwards the sand behaviour evaluated in the Lagrangian frame using the DPM model of ANSYS Fluent. The solid phase was analysed and characterise through two main parameters: Stokes number and volume fraction. For a better understanding of the interaction between sand and train, external routines were implement in Fluent to extend the capabilities of the DPM model. These routines allow for the measurement of the force exerted by the particles on the train and the representation of the distribution of impacts in the surface of the ETR500 model.

The results conclude that the forward head is the region that receives most of the impacts, being therefore more susceptible to the wear of the materials. Collisions are noticed as well in the roof of the train, especially in the last cars. The probability of impact grows with Stokes number, because the particles are less affected by the fluid stream. On the other hand, the force exchange has been properly measured. The graphs indicate an increase in the drag coefficient, which is higher for larger Stokes number and volume fraction. For the range of particles testes, the maximum drag coefficient ratio corresponds to 1.11.

Further work will extend the current investigation, evaluating a broader range of particle characteristics and injection locations.

### REFERENCES


Computation of Two-Point Contact between Wheelset and Rail Exploiting the Structure of Non-Penetrating Contact Equations for Straight Track

Behrooz Fallahi* and Chao Pan#

* Faculty of Mechanical Engineering
Northern Illinois University
590 Garden Road, Dekalb 60115
bfallahi@niu.edu

# Graduate Student
Northern Illinois University
590 Garden Road, Dekalb 60115
z1647714@students.niu.edu

ABSTRACT

Computation of the contact points between wheel and rail is a fundamental problem in railroad vehicle modeling. In solving the equations for the contact points, two questions should be answered. They are: 1) what is the number of contact points? And 2) where are their locations? To answer these questions, the non-penetrating contact constraints are derived in track coordinate system. It is shown that five equations per contact should be constructed. Four of these equations locate common normal (referred to as common-normal constraints) and the fifth equation makes the length of common normal to become zero (this equation is referred to as distance equation). The common-normal constraints as expressed in track coordinate system for straight rail have a special structure. In the first equation only two unknowns exist. In the subsequent equations, each equation has one more unknown than the previous one. This feature is exploited to devise an algorithm for computation of the two-contact-point problem.

Keywords: Railroad vehicle simulation, Common normal, Contact point

INTRODUCTION

In dynamic simulation of railroad vehicle two approaches are used to incorporate the contact between rail and wheelset [1-6]. They are elastic and embedded constraint formulations. In embedded approach, the wheel surface is tangent to the rail, Shabana and Berzeri [1]. In elastic contact approach, a small penetration of the wheel surface into rail surface is allowed [2]. Yamashita and Sugiyama [5] used elastic contact approach to compute the flange contact while the tread contacts are computed using constraint contact approach.

To model the dynamics of a railroad vehicle, it is of paramount importance to detect all contacts. As without it, the dynamic simulation for study of wheel and rail interaction, derailment, curving, and moving over a switch/turnout, is not correct. Shabana et al reported nodal search to detect multiple contacts [3]. In the nodal search, several nodes are selected on wheel and rail profile. The distance between every node pair is computed and the intervals within which the distance decreases and then increases are identified. These intervals contain a common normal.

Yamashita et al implemented flange climb simulation [5]. In their study, they used embedded constraint formulation for the tread contacts and elastic formulation for the flange contact. Lagrange multipliers are used to detect the onset of loss of tread contact. After the loss of the tread contact, the elastic contact formulation for the flange is switched to embedded-constraint contact formulation.

Auciello et al. [7] developed a specialized technique to compute the common normal for straight track. They reported that the method is very fast.
Fallahi and Sunil [8] introduced the concept of signed distance function and used it to compute the number of common normal and an estimate of their locations. They reported that their method is very slow but it has the advantage of observing the location of the common normal through plotting the contours of signed distance function. This can be used for verification of their locations.

Perla [9] developed a procedure to compute common normal between a pair of wheelset and rails. In his study, the concept of “shadow rail” was introduced. To construct the shadow rail, the intersection of the vertical plane that passes through the wheel axis and the curve track was computed. The intersection point was called proximity point. Next, the tangent line at this intersection point was constructed. The shadow rail was constructed by sweeping the rail profile along this tangent line. The shadow rail is a good approximation of the rail near the proximity point. It is shown that the four equations governing the location of the common normal between the wheelset and the shadow rail has a special structure. This special structure was exploited to devise a more efficient algorithm for computing common normal. In this study, Perla’s approach was extended to compute two-point contact between a wheelset and the rails.

FORMULATION OF EQUATIONS FOR THE CHORD, TANGENTS, AND NORMALS

Recall that the position vector of a point on the rail surface in absolute coordinate system is [9]:

\[ r_p^P = R^e + A^e r_p^P \]

Assuming that the track coordinate system is aligned with the absolute coordinate system, then,

\[ R^e = [s_{tr}, 0, 0]^T \]

Wherever \( s_{tr} \) is the arc length along the track measured from the origin of the track coordinate system. As will be shown in this section, it is advantageous to express the parametric equations of the rail and wheelset surface in the track coordinate system. This leads to a system of equations whose structure will be exploited to devise a more efficient computational scheme. Following this, the parametric equation of the rail in the track coordinate system is:

\[ r_{tr}^P = [0, y_{tr}^P, z_{tr}^P(\gamma_{tr}^P)]^T \] (1)

Note that the rail surface parameters are \( s_{tr} \) and \( y_{tr}^P \). Therefore the two tangents to the rail surface are:

\[ (t_1)_{tr}^P = \frac{d}{dy_{tr}}(r_{tr}^P) = \begin{bmatrix} 0 & 1 & d/dy_{tr}(z_{tr}^P(\gamma_{tr}^P)) \end{bmatrix}^T \] (2)

\[ (t_2)_{tr}^P = \frac{d}{ds_{tr}}(r_{tr}^P) = [1 \ 0 \ 0]^T \] (3)

Forming the cross product of these two tangents, the normal to rail at point P can be obtained:

\[ n_{tr}^P = (t_1)_{tr}^P \times (t_2)_{tr}^P = \begin{bmatrix} 0 & d/dy_{tr}(z_{tr}^P(\gamma_{tr}^P)) & -1 \end{bmatrix}^T \] (4)

Using Perla’s formulation [9], the parametric equation of the wheelset surface in the track coordinate system is:

\[ r_{tr}^{ws} = U + W(R_{ctr}^{ws} + A_{ctr}^{ws} \cdot (r_c + R_{y}(\alpha r_{wp}))) \] (5)
where,

\[
W = \left( A^v \right)^T A_{\text{ctr}}^{\text{ws}}
\]

\[
U = \left( A^v \right)^T (R_{\text{ctr}} - R^r)
\]

The surface parameters for the wheelset surface are \( \alpha \) and \( y_{wp} \). Therefore, the equations for the tangents to the wheelset surface are:

\[
(t^p_{1 \text{tr}})^{\text{ws}} = W \cdot A_{\text{ctr}}^{\text{ws}} \cdot R_y(\alpha) \cdot \begin{bmatrix} 0 & 1 & \frac{d}{dy_{wp}} z^p_{wp}(y_{wp}) \end{bmatrix}^T
\]

\[
(t^p_{2 \text{tr}})^{\text{ws}} = W A_{\text{ctr}}^{\text{ws}} \frac{d}{d\alpha} (R_y(\alpha)) r^p
\]

The expression for the normal to wheelset surface is:

\[
(n^{\text{ws}}_{\text{tr}})^n_{\text{tr}} = (t^p_{1 \text{tr}})^{\text{ws}} \times (t^p_{2 \text{tr}})^{\text{ws}}
\]

The expression for the chord (the vector connecting a pair points on wheelset and rail surface) in the track coordinate system is, \( d_{\text{tr}} \):

\[
d_{\text{tr}} = r^p_{\text{tr}} - r^p
\]

\( r^p_{\text{tr}} \) and \( r^p \) are given by Equation (5) and (1), respectively.

**FORMULATION OF THE NON-PENETRATING EQUATIONS FOR THE CONTACT POINTS**

To enforce contact between the wheelset and the rail, five equations per contact should be constructed. In this study the five equations are grouped into two sets. The first set, which consists of four equations, locates the common normal. The second set which includes one equation enforces the length of the common normal to become zero. The unknown parameters are four surface parameters per contact, the yaw and roll angle, \( y\)- and \( z\)-coordinate of the wheelset coordinate system relative to the center track coordinate system, and the arc length of the track coordinate system. The advantage of this formulation is that only the parameters of wheelset and rail surface appear in the common normal constraints. This structure is exploited in this study to devise a more efficient Newton iterate.

In the following, the formulation of the five equations for the non-penetrating condition is presented. The first equation is formed by requiring the projection of the chord \( d_{\text{tr}} \) onto the normal \( n^p_{\text{tr}} \) to be zero. That is:

\[
g = d_{\text{tr}} \cdot n^p_{\text{tr}} = 0
\]

Substitute Equation (5) and (9) into Equation (10) to get:

\[
g = (u_2 + w_{2,2} y_{\text{ctr}}^{\alpha} + w_{2,3} y_{\text{ws}}^{\alpha} + \frac{1}{2} (-w_{2,1} s\phi c\varphi_{\text{ctr}}^{\alpha} + w_{2,2} c\psi c\varphi_{\text{ctr}}^{\alpha} + w_{2,3} s\phi) G + (-w_{2,1} s\psi c\varphi_{\text{ctr}}^{\alpha} + w_{2,2} c\psi c\varphi_{\text{ctr}}^{\alpha} + w_{2,3} s\phi) y_{\text{wp}} + (-w_{2,1} c\psi_{\text{ctr}}^{\alpha} + w_{2,2} c\varphi_{\text{ctr}}^{\alpha} + w_{2,3} c\psi_{\text{ctr}}^{\alpha}) c\alpha z_{\text{wp}} - y_{\text{rp}}) \cdot dz_{\text{rp}} - (u_3 + w_{3,2} y + w_{3,3} z + \frac{1}{2} (-w_{3,1} s\psi c\varphi_{\text{ctr}}^{\alpha} + w_{3,2} c\psi c\varphi_{\text{ctr}}^{\alpha} + w_{3,3} s\psi_{\text{ctr}}^{\alpha}) G + (-w_{3,1} s\psi c\varphi_{\text{ctr}}^{\alpha} + w_{3,2} c\varphi_{\text{ctr}}^{\alpha} + w_{3,3} s\psi_{\text{ctr}}^{\alpha}) y_{\text{wp}} + (-w_{3,1} c\varphi_{\text{ctr}}^{\alpha} + w_{3,2} c\varphi_{\text{ctr}}^{\alpha}) c\alpha z_{\text{wp}} - z_{\text{rp}}) = 0
\]
where $c(*) = \cos(*)$ and $s(*) = \sin(*)$.

Since the normal to the rail and normal to the wheelset are parallel, the ratio of their $y$-component to the $z$-component must be equal that is:

$$f_2 = \left[ w_{2,1}^p\psi_{1,1} + w_{2,2}^p\psi_{1,2} \cos(\alpha) + w_{2,3}^p\psi_{1,3} \sin(\alpha) \right] - \left[ (w_{2,1}^p\psi_{1,1} + w_{2,2}^p\psi_{1,2} \cos(\alpha) + w_{2,3}^p\psi_{1,3} \sin(\alpha)) d_{y_{wp}}(z_{wp}(y_{wp})) \right] = 0$$

Since the $x$-component of the normal to the rail is zero and the chord and the normal to the rail are parallel, the $x$-component of the chord normal to the rail must also be zero. That is:

$$f_3 = \left[ u_1 + w_{1,2}^y y + w_{1,3}^z \right] + \frac{1}{2} (-w_{3,1}^\phi \psi_{1,1} \cos(\alpha) + w_{1,2}^\psi \psi_{1,2} \sin(\alpha) + w_{1,3}^\psi \psi_{1,3} + w_{1,3}^y G) + \left[ (w_{3,1}^\phi \psi_{1,1} + w_{1,2}^\psi \psi_{1,2} \sin(\alpha) + w_{1,3}^\psi \psi_{1,3} \cos(\alpha) \right] z_{wp}(y_{wp}) - s\alpha_{tr} = 0$$

Since the normal and the chord are parallel, the ratio of the $y$-component to $z$-component of the chord normal to the rail must be equal. That is:

$$f_4 = \left[ w_{3,1}^\psi \psi_{1,1} + w_{2,2}^\psi \psi_{1,2} \sin(\alpha) - (w_{3,1}^\phi \psi_{1,1} \cos(\alpha) - w_{3,2}^\psi \psi_{1,2} \sin(\alpha) + w_{1,3}^\psi \psi_{1,3} \cos(\alpha) \right] z_{wp} (y_{wp}) - \left[ u_2 + w_{1,2}^\psi \psi_{1,2} + w_{1,3}^z \right] + \frac{1}{2} (-w_{3,1}^\phi \psi_{1,1} \cos(\alpha) + w_{1,2}^\psi \psi_{1,2} \sin(\alpha) + w_{1,3}^\psi \psi_{1,3} \cos(\alpha) + w_{1,3}^y G) + \frac{d_{y_{wp}}}{d_{y_{wp}}(z_{wp}(y_{wp}))} + \left[ (w_{2,1}^\psi \psi_{1,1} + w_{2,2}^\psi \psi_{1,2} \sin(\alpha) - w_{2,2}^\psi \psi_{1,2} \sin(\alpha) + w_{2,3}^\psi \psi_{1,3} \cos(\alpha) \right] z_{wp}(y_{wp}) - \left[ u_2 + w_{1,2}^\psi \psi_{1,2} + w_{1,3}^z \right] + \frac{1}{2} (-w_{2,1}^\phi \psi_{1,1} \cos(\alpha) + w_{2,2}^\psi \psi_{1,2} \sin(\alpha) + w_{2,3}^\psi \psi_{1,3} \cos(\alpha) + w_{2,3}^y G) - (w_{2,1}^\psi \psi_{1,1} \cos(\alpha) + w_{2,2}^\psi \psi_{1,2} \sin(\alpha) + w_{2,3}^\psi \psi_{1,3} \cos(\alpha) + w_{2,3}^y G) + y_{wp} = 0$$
Equations (11) through (15) are the non-penetrating equation for each contact point.

**TWO-POINT CONTACT PROBLEM**

In two-points-contact problem, there are two tread contacts (one on each side). In the following development, the set of independent parameters is: \(s_{ctt}, y_{w}^{ws}, \) and \(\psi_{w}^{ws}.\) The dependent parameters for three-point contact problem are: \(z_{ctt}^{ws}, \phi_{ctt}^{ws}, \alpha, y_{w}, \) (wheel surface parameters), and \(y_{r}, s_{tr} \) (rail surface parameters); for each contact. Therefore the Jacobean matrix is:

\[
C = \begin{bmatrix}
\frac{\partial f_1}{\partial \alpha} & \frac{\partial f_1}{\partial y_{w}} & 0 & 0 & 0 & \frac{\partial f_1}{\partial \psi_{ctt}^{ws}} \\
\frac{\partial f_2}{\partial \alpha} & \frac{\partial f_2}{\partial y_{w}} & 0 & 0 & 0 & \frac{\partial f_2}{\partial \psi_{ctt}^{ws}} \\
\frac{\partial f_3}{\partial \alpha} & \frac{\partial f_3}{\partial y_{w}} & \frac{\partial f_3}{\partial \psi_{ctt}^{ws}} & \frac{\partial f_3}{\partial \psi_{ctt}^{ws}} & \frac{\partial f_3}{\partial \alpha} & \frac{\partial f_3}{\partial \psi_{ctt}^{ws}} \\
\frac{\partial f_4}{\partial \alpha} & \frac{\partial f_4}{\partial y_{w}} & \frac{\partial f_4}{\partial \psi_{ctt}^{ws}} & \frac{\partial f_4}{\partial \psi_{ctt}^{ws}} & \frac{\partial f_4}{\partial \alpha} & 0 \\
\frac{\partial f_5}{\partial \alpha} & \frac{\partial f_5}{\partial y_{w}} & \frac{\partial f_5}{\partial \psi_{ctt}^{ws}} & \frac{\partial f_5}{\partial \psi_{ctt}^{ws}} & 0 & \frac{\partial f_5}{\partial \alpha} \\
\frac{\partial f_6}{\partial \alpha} & \frac{\partial f_6}{\partial y_{w}} & \frac{\partial f_6}{\partial \psi_{ctt}^{ws}} & \frac{\partial f_6}{\partial \psi_{ctt}^{ws}} & 0 & 0 \\
\frac{\partial f_7}{\partial \alpha} & \frac{\partial f_7}{\partial y_{w}} & \frac{\partial f_7}{\partial \psi_{ctt}^{ws}} & \frac{\partial f_7}{\partial \psi_{ctt}^{ws}} & 0 & 0
\end{bmatrix}
\]

(16)

Partition the Jacobean matrix to get:

\[
C_{11} = \begin{bmatrix}
\frac{\partial f_1}{\partial \alpha} & \frac{\partial f_1}{\partial y_{w}} & 0 \\
\frac{\partial f_2}{\partial \alpha} & \frac{\partial f_2}{\partial y_{w}} & 0 \\
\frac{\partial f_3}{\partial \alpha} & \frac{\partial f_3}{\partial y_{w}} & \frac{\partial f_3}{\partial \psi_{ctt}^{ws}} \\
\frac{\partial f_4}{\partial \alpha} & \frac{\partial f_4}{\partial y_{w}} & \frac{\partial f_4}{\partial \psi_{ctt}^{ws}} \\
\frac{\partial f_5}{\partial \alpha} & \frac{\partial f_5}{\partial y_{w}} & \frac{\partial f_5}{\partial \psi_{ctt}^{ws}} \\
\frac{\partial f_6}{\partial \alpha} & \frac{\partial f_6}{\partial y_{w}} & \frac{\partial f_6}{\partial \psi_{ctt}^{ws}} \\
\frac{\partial f_7}{\partial \alpha} & \frac{\partial f_7}{\partial y_{w}} & \frac{\partial f_7}{\partial \psi_{ctt}^{ws}}
\end{bmatrix}
\]

\[
C_{13} = \begin{bmatrix}
0 & \frac{\partial f_1}{\partial \psi_{ctt}^{ws}} \\
0 & \frac{\partial f_2}{\partial \psi_{ctt}^{ws}} \\
\frac{\partial f_3}{\partial \psi_{ctt}^{ws}} & \frac{\partial f_3}{\partial \psi_{ctt}^{ws}} \\
\frac{\partial f_4}{\partial \psi_{ctt}^{ws}} & \frac{\partial f_4}{\partial \psi_{ctt}^{ws}} \\
\frac{\partial f_5}{\partial \psi_{ctt}^{ws}} & \frac{\partial f_5}{\partial \psi_{ctt}^{ws}} \\
\frac{\partial f_6}{\partial \psi_{ctt}^{ws}} & \frac{\partial f_6}{\partial \psi_{ctt}^{ws}} \\
\frac{\partial f_7}{\partial \psi_{ctt}^{ws}} & \frac{\partial f_7}{\partial \psi_{ctt}^{ws}}
\end{bmatrix}
\]

\[
C_{21} = \begin{bmatrix}
\frac{\partial g}{\partial \alpha} & \frac{\partial g}{\partial y_{w}} & \frac{\partial g}{\partial y_{w}}
\end{bmatrix}
\]

\[
C_{23} = \begin{bmatrix}
\frac{\partial g}{\partial \psi_{ctt}^{ws}} & \frac{\partial g}{\partial \psi_{ctt}^{ws}}
\end{bmatrix}
\]

\[
C_{31} = \begin{bmatrix}
\frac{\partial f_1}{\partial \alpha} & \frac{\partial f_1}{\partial y_{w}} & 0
\end{bmatrix}
\]

\[
C_{32} = \begin{bmatrix}
\frac{\partial f_1}{\partial \psi_{ctt}^{ws}} & \frac{\partial f_1}{\partial \psi_{ctt}^{ws}}
\end{bmatrix}
\]

\[
C_{33} = \begin{bmatrix}
\frac{\partial f_1}{\partial \psi_{ctt}^{ws}} & \frac{\partial f_1}{\partial \psi_{ctt}^{ws}}
\end{bmatrix}
\]

The correction vector \(\Delta X\) is:

\[
\Delta X = [\Delta \alpha, \Delta y_{w}, \Delta y_{r}, \Delta s_{tr}, \Delta z_{ctt}^{ws}, \Delta \phi_{ctt}^{ws}]^{T}
\]

and it is partitioned as:

\[
\Delta X_{1} = [\Delta \alpha, \Delta y_{w}, \Delta y_{r}]^{T}, \quad \Delta X_{2} = \Delta s_{tr}, \quad \Delta X_{3} = [\Delta z_{ctt}^{ws}, \Delta \phi_{ctt}^{ws}]^{T}
\]

The right-hand side vector is:

\[
V = [V_{1}, V_{2}, V_{3}]^{T}
\]

It is partitioned as:

\[
V_{1} = [f_{1}, f_{2}, f_{4}]^{T}, \quad V_{2} = g, \quad V_{3} = f_{3}
\]

Note that since there are two contact points, two equations of the form,

\[
[c_{21} - c_{23}(c_{11})^{-1}c_{13}] \Delta x_{3} = v_{3} - c_{23}(c_{11})^{-1}v_{1}
\]

can be constructed. That is:
\[
\begin{bmatrix}
C_{23} - C_{21}(C_{11})^{-1}C_{13} \\
C_{23} - C_{21}(C_{11})^{-1}C_{13}
\end{bmatrix} \Delta x_3 = \begin{bmatrix}
V_{1} - C_{21}(C_{11})^{-1}V_1 \\
V_{2} - C_{21}(C_{11})^{-1}V_2
\end{bmatrix}
\] (18)

Note that superscript in Equation (18) refers to contact point number. Equation (18) can be solved for \(\Delta X_3\). Next, \(\Delta X_1\) can be computed from,

\[
\Delta X_1 = C_{11}^{-1} \cdot (V_1 - C_{13} \cdot \Delta X_3)
\] (19)

and \(\Delta X_2\) can be computed using Equation (3.28). That is:

\[
C_{32}\Delta x_2 = V_3 - C_{31}\Delta x_1 - C_{33}\Delta x_3
\] (20)

Based on equations derived in this section, the following algorithm is used to compute the correction terms:

Step 1: Set the initial value of the unknown vector \([\alpha, y_{wp}, y_{rp}, s_{tr}, Z_{ctr}, \varphi_{ctr}]^T\)

Step 2: Compute the Vector function \(V\)

Step 3: Compute the Jacobean matrix \(C\) and form its partitions \(C_{11}, C_{13}, C_{21}, C_{23}, C_{31}, C_{32},\) and \(C_{33}\).

Step 4: Use Equation (18) and compute \(\Delta x_3\).

Step 5: Use Equation (19) and compute \(\Delta x_1\).

Step 6: Use Equation (20) and compute \(\Delta x_2\)

\(\Delta x_1, \Delta x_2,\) and \(\Delta x_3\) are used in a Newton iterate to compute the contact point locations. This completes the procedure for computing the correction terms for three-contact-points problem.

RESULTS FOR TWO CONTACT-POINT

To verify the procedure presented, four positions and orientations of the wheelset are arbitrary selected, see Table 1.

<table>
<thead>
<tr>
<th>Case</th>
<th>(s_{ctr})</th>
<th>(y_{ctr})</th>
<th>(s_{ctr}^{ws})</th>
<th>(y_{ctr}^{ws})</th>
<th>(\psi_{ctr}^{ws})</th>
<th>(\psi_{ctr}^{ws})</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.2</td>
<td>0.02</td>
<td>0.433</td>
<td>0</td>
<td>7(^\circ)</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0.2</td>
<td>0.01</td>
<td>0.419</td>
<td>0</td>
<td>7(^\circ)</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0.2</td>
<td>-0.01</td>
<td>0.419</td>
<td>0</td>
<td>7(^\circ)</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0.2</td>
<td>-0.02</td>
<td>0.423</td>
<td>0</td>
<td>7(^\circ)</td>
<td></td>
</tr>
</tbody>
</table>

For the positions and orientations of the wheelset reported in Table 1, Perla’s procedure is used and the locations of common normal are computed. Two of the common normals with shortest length are selected. Their parameters are used as the initial guess for the two-point contact problem and the algorithm of this work is used to compute the contact points. The unknowns: \(z_{ctr}, \varphi_{ctr}^{ws}, \alpha, y_{wp}\) (wheel surface parameters), and \(y_{rp}, s_{ctr}\) (rail surface parameters); for each contact are reported. They are reported in Table 2 through 4.

<table>
<thead>
<tr>
<th>Case</th>
<th>(z_{ctr}^{ws})</th>
<th>(\varphi_{ctr}^{ws})</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.850</td>
<td>-56.608(^\circ)</td>
</tr>
<tr>
<td>2</td>
<td>0.370</td>
<td>-0.992(^\circ)</td>
</tr>
<tr>
<td>3</td>
<td>0.368</td>
<td>1.372(^\circ)</td>
</tr>
<tr>
<td>4</td>
<td>0.370</td>
<td>2.066(^\circ)</td>
</tr>
</tbody>
</table>
Table 3: Surface parameters for contact point on left track

<table>
<thead>
<tr>
<th>Case</th>
<th>$y_{wp}$</th>
<th>$\alpha$</th>
<th>$y_{re}$</th>
<th>$s_{re}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-0.026</td>
<td>176.831°</td>
<td>-0.016</td>
<td>0.119</td>
</tr>
<tr>
<td></td>
<td>-0.032</td>
<td>175.949°</td>
<td>0.031</td>
<td>0.166</td>
</tr>
<tr>
<td>2</td>
<td>-0.004</td>
<td>179.946°</td>
<td>-0.001</td>
<td>0.109</td>
</tr>
<tr>
<td>3</td>
<td>0.001</td>
<td>179.656°</td>
<td>-0.007</td>
<td>0.109</td>
</tr>
<tr>
<td>4</td>
<td>0.005</td>
<td>179.571°</td>
<td>-0.008</td>
<td>0.108</td>
</tr>
</tbody>
</table>

Table 4: Surface parameters for contact point on right side

<table>
<thead>
<tr>
<th>Case</th>
<th>$y_{wp}$</th>
<th>$\alpha$</th>
<th>$y_{re}$</th>
<th>$s_{re}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>-0.005</td>
<td>180.298°</td>
<td>0.006</td>
<td>0.291</td>
</tr>
<tr>
<td>3</td>
<td>-0.007</td>
<td>180.007°</td>
<td>0.001</td>
<td>0.291</td>
</tr>
<tr>
<td>4</td>
<td>-0.003</td>
<td>179.992°</td>
<td>-0.002</td>
<td>0.290</td>
</tr>
</tbody>
</table>

To verify the solutions obtain in this study, the procedure developed by Perla [9] is used to plot $f_4$ vs. $y_{wp}$. For all cases, the zero crossing of $f_4$ matched the contact points found by the algorithm of this study. Figure 2 and 3 shows such plots for case 2.

Figure 2: Plot of $f_4$ for case 2 (left track)

Figure 3: Plot of $f_4$ for case 2 (right track)
CONCLUSION

In this study an approach to enforce two contacts between a wheelset and rails are developed. This is accomplished by developing parametric equation of wheelset and rail as revolve and sweep surface. Then the expression for the tangent, normal, and the chord vector in track coordinate system is developed. It is shown that using these expressions, the Jacobean of non-penetrating equations has a block structure. These structure is exploited to devise a Newton iterate that does not use the full Jacobean matrix. For cases are solved using this algorithm and the solutions are verified.

REFERENCES


Computational Model Updating for a Rail Vehicle Bogie System of a High-Speed Train Based on Combined Component Analysis

Qintao Guo*, Yanhe Tao*, Haitao Li#, Dandan Yang*, Long Ma#, Baoqiang Zhang*, Tong Wang*, Lingmi Zhang*

* Department of College of mechanical and electrical engineering, Nanjing University of Aeronautics & Astronautics
29 Street Yudao, 210016, Nanjing, China

#CSR Qingdao Sifang Locomotive & Rolling Stock Co., Ltd.
No.88 in Jinhong east road of Chengyang district
266000, Qingdao, China

guo_qintao@nuaa.edu.cn
lihaitao@cqsf.com

ABSTRACT
Computational simulation of dynamic behavior of a complex mechanical structure is of profound importance in structural dynamic analysis. Computational model updating of a rail vehicle bogie system of a high speed train based on combined structural analysis is presented in this paper. In order to validate the models with reliable and accurate dynamic characteristics of the components, Computational model updating of the components are conducted based on model correlation and sensitivity analysis. Pre-updating FE models of the two major components are built, and the appropriate parameters are selected for the following parametric model updating.

Keywords: vehicle dynamics, computational model updating, experimental modal analysis, high-speed train, component analysis

1. INTRODUCTION
Computational simulation of dynamic behavior of a complex mechanical structure, such as rail vehicle bogie is of profound importance in structural dynamic analysis and design of high-speed train.

Computational model updating\(^1\) of a rail vehicle bogie system of a high speed train based on combined structural analysis is presented in this paper. Computational models of the major components, i.e. frame and wheel set were established via FEM at first. In order to validate the models with reliable and accurate dynamic characteristics of the components, advanced experimental modal analysis (EMA) technique based on multiple instrumented hammer excitations is utilized. Modal parameters identified from EMA are validated via variety of measures, such as reciprocity check via cross FRFs, FRF curve fittings, etc. Computational model updating (CMU) of the components are conducted based on model correlation and sensitivity analysis. Pre-updating FE models\(^2\) of the two major components are built, and the appropriate parameters are selected for the following parametric model updating.

2. BASIC THEORY-ALGORITHM IN UPDATING TECHNIQUE
Sensitivity analysis based on modal eigenvalues and/or eigenvectors is a main method used in solving structural dynamics model updating problems with large-scale degree of freedom. The basic theory of the method has developed maturely. The main steps and the calculation formula is as follows.

Known that the FEM of structure consists of n design parameters, including former m for updating, then design parameters can be expressed as:
The global stiffness and mass matrices of the structure can be expressed in the design parameter functions:

\[ K = f_K(p) \quad M = f_M(p) \]  \hspace{1cm} (1.2)

The corresponding characteristics can be shown as the function of design parameters:

\[ f = F(K, M) = F(f_K(p), f_M(p)) = f_p(p) \]  \hspace{1cm} (1.3)

where \( f \) can be any characteristic, such as modal frequency, modal shape, etc., or a combination of them.

The problem of model updating is transformed to the following optimization problem:

\[ \min \left\| W_p \cdot R(p) \right\|^2, R(p) = \{ f_e \} - \{ f_p(p) \} \]
\[ \text{s.t.} \quad \text{VLB} \leq p \leq \text{VUB} \]  \hspace{1cm} (1.4)

where \( f_e \) and \( f_p(p) \) representing the experimental and analysis values of the structural characteristic, \( R(p) \) is called the residuals, while \( \text{VLB} \) and \( \text{VUB} \) representing lower the and upper bounds of the structural design parameters. \( W_p \) representing the weighted matrix between various structure characteristic parameters.

Typically, \( f_p(p) \) is a nonlinear function of design parameters, which can be expanded into the first order Taylor expression of the updating parameters to convert the nonlinear problem into a linear problem.

\[ f_p(p) = f_p(p_0) + S \Delta p \]  \hspace{1cm} (1.5)

Where \( p_0 \) is the initial value of design parameter.

\[ S = \frac{\partial f_p}{\partial p} \bigg|_{p=p_0} \]  \hspace{1cm} (1.6)

The above expression of \( S \) represents sensitivity matrix of eigenvalues with respect to design parameters, \( \Delta p = p - p_0 \) represents the error of design parameters.

Sensitivity of eigenvalues with respect to parameters is

\[ \frac{\partial \lambda_i}{\partial p} \bigg|_{p=p_0} = \varphi_i \left[ -\lambda_i \frac{\partial M}{\partial p} + \frac{\partial K}{\partial p} \right] \varphi_j \]  \hspace{1cm} (1.7)

There is no analytical expression to calculate the sensitivity of eigenvectors respect to design parameters, the commonly used methods are Nelson method, classical modal method, modified modal method, iterative modal method and complete modal method. Nelson method is most common in the methods mentioned above and a lot of finite element software such as Nastran calculate the sensitivity of vibration mode utilizing this method.

The sensitivity of MAC (Modal Assurance Criterion) can be calculated by the sensitivity of eigenvectors.[2] The sensitivity of MAC between the \( i \)-th order test mode \( \varphi_i^e \) and the \( j \)-th order calculated mode \( \varphi_j^a \), respect to the \( n \)-th updating parameters \( p_n \) could be calculated according to the following formula

\[ S = \frac{\partial f_p}{\partial p} \bigg|_{p=p_0} \]
\[
\frac{\partial \text{MAC}_{ij}}{\partial p_n} = \frac{2 \left( \{\phi_j^r\}^T \{\phi_j^a\} \right) \{\phi_j^a\}^T \frac{\partial \phi_j^a}{\partial p_n} \{\phi_j^a\}^T}{\left( \{\phi_j^r\}^T \{\phi_j^r\} \right) \left( \{\phi_j^a\}^T \{\phi_j^a\} \right)} - \frac{2 \left( \{\phi_j^r\}^T \{\phi_j^a\} \right)^2 \{\phi_j^a\}^T \frac{\partial \phi_j^a}{\partial p_n} \{\phi_j^a\}^T}{\left( \{\phi_j^r\}^T \{\phi_j^r\} \right)^2}
\]  

(1.8)

Where \( \frac{\partial \phi_j^a}{\partial p_n} \) is the partial derivative of \( j \)th order calculated mode \( \phi_j^a \) to design parameter \( p_n \). The diagonal elements of the MAC matrix is large, the non-diagonal elements is relatively small, and the target of analysis is 0. Generally, in order to improve the performance of sensitivity matrix, the diagonal elements after the mode pairing, i.e. \( \frac{\partial \text{MAC}_{ij}}{\partial p_n} \), are adopted to in the model updating.

The extreme-value problem is converted to a linear problem by using the Lagrange multiplier method as follows:

\[
W_j \Delta p = W_j (f_c - f_p(p_n))
\]  

(1.9)

Formula 1.9 is a common model updating equation, which is an iterative optimization process. There are advisable results in general case. The regularization is necessary when the condition number of the weighted matrix and the sensitivity matrix is poor. When the number of parameters is larger, it is necessary to use optimization search method. In this paper, the optimization method which uses SQP (Sequential Quadratic Programming) to search trust region is adopted.

3. UPDATING OF THE COMPONENTS AND A HALF BOGIES SYSTEM

The bogie structure system is the key of system dynamics of high speed train. An accurately modeled bogie structure system is crucial in both analysis of the acceleration vibration response and improvement of the dynamic stress of structure strength. According to the characteristics of the structure system of a high speed train bogie that the whole system can be divided into three parts including wheel sub structure(down support), frame sub structure and main body which instead by simulation of the rigid mass block. In order to obtain the model with high precision, the priority of structures or sub structures which need to be updated is frame, then wheel sub structure, and then bogie system in overall condition(By adding simulation of car body).

3.1 Finite Element Model Updating of Frame sub Structure

Figure 3.2 shows the 3D model of overall frame structure established by FEA software. Eigenvalues and eigenvectors can be extracted based on the Lanczos method which belongs to modal analysis module of a finite element software.

Pre-updating model and Parameter chosen

The initial finite element model can not be directly applied to model updating, so the first step is to obtain the pre-updating model for the subsequent process of model updating. The bottom of motor and frame are connected by bolts which are simulated by beam elements (Partial Enlargement as Figure 3.1), and the joint stiffness of beams are parameters should be calibrated. The pre-updating model consists of 35944 nodes and 46380 units.
The calculation result of MAC

The number of modals calculated by finite element simulation is far more than those captured in test, which indicates that test just can achieve only a portion of structural modals. Here, three translational degrees of freedom \((x, y, z)\) are selected to calculate MAC. Table 3.1 shows the main 3 modal frequencies and MAC error before and after updating.

**Table 3.1** comparison of the main 3 modal frequencies and MAC error that before and after the updating of frame structure

<table>
<thead>
<tr>
<th>Test order</th>
<th>Test frequency (Hz)</th>
<th>Finite element order</th>
<th>Finite element frequency (Hz)</th>
<th>MAC before correction</th>
<th>Frequency error before correction(%)</th>
<th>MAC after correction</th>
<th>Frequency error after correction (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>41.75</td>
<td>1</td>
<td>42.3672</td>
<td>0.92</td>
<td>-7.1900</td>
<td>0.96</td>
<td>-1.4783</td>
</tr>
<tr>
<td>2</td>
<td>55.39</td>
<td>2</td>
<td>56.1488</td>
<td>0.86</td>
<td>-5.7824</td>
<td>0.85</td>
<td>-1.3699</td>
</tr>
<tr>
<td>3</td>
<td>70.01</td>
<td>3</td>
<td>63.8302</td>
<td>0.90</td>
<td>-15.7356</td>
<td>0.91</td>
<td>8.8270</td>
</tr>
<tr>
<td>mean absolute value</td>
<td></td>
<td></td>
<td></td>
<td>0.89</td>
<td>9.5693</td>
<td>0.91</td>
<td>3.8917</td>
</tr>
</tbody>
</table>

3.2 Correction Results of The Wheel sub Structure

Pre-updating model and Correction parameter selection. The pre-updating model consists of 34819 nodes and 22948 elements. In order to obtain the model corresponding to experimental test, lumped mass as the mass of vibration of bearing, gear shaft and axle box is appended, as is shown in Figure 3.4 Partial Enlargement of lumped mass. Using BUSH element to replace the contact and connection bolts between the wheel and the brake disc in the pre-updating model, at the same time that the BUSH element stiffness as parameters to be calibrated. In consideration of the material differences between wheel felloe, Internal hub and flexible shaft and the simplifies during finite element modeling, the material properties of the three parts are described by three different material properties.
Table 3.2 Comparison of the MAC and frequency error that before and after the correction of the finite element model

<table>
<thead>
<tr>
<th>Test order</th>
<th>Test frequency (Hz)</th>
<th>Finite element order</th>
<th>Finite element frequency (Hz)</th>
<th>MAC before correction</th>
<th>Frequency error before correction (%)</th>
<th>MAC after correction</th>
<th>Frequency error after correction (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>79.7</td>
<td>2</td>
<td>74.9</td>
<td>0.86</td>
<td>-6.02</td>
<td>0.86</td>
<td>4.27</td>
</tr>
<tr>
<td>3</td>
<td>87.5</td>
<td>1</td>
<td>74.2</td>
<td>0.71</td>
<td>-15.20</td>
<td>0.72</td>
<td>-6.01</td>
</tr>
<tr>
<td>4</td>
<td>184.1</td>
<td>5</td>
<td>167.0</td>
<td>0.88</td>
<td>-9.29</td>
<td>0.89</td>
<td>-4.87</td>
</tr>
<tr>
<td>mean absolute value</td>
<td>0.86</td>
<td></td>
<td></td>
<td>10.17</td>
<td></td>
<td>0.87</td>
<td>3.87</td>
</tr>
</tbody>
</table>

As can be seen from the table above, the initial finite element model of the wheel structure has high precision, frequency error of the absolute value of mean value is 10.17%, MAC is 0.86.

4. RESIDUAL ITERATION RESULTS OF THE TARGET FREQUENCY

The finite element model and the correction parameter selection. The FE(finite element) model of the overall bogie is meshed by the finite element software, which consists of 119994 nodes and 116262 elements. The mass of the car-body model is 32 tons, the mass of the bogie model is 7 tons. The axle box vertical damper, the axial location nodes, air spring, etc. are simulated by BUSH element. In most cases, the updating parameters of this model are connection stiffness. Finally, six order of frequencies are selected as the target of model updating and the six parameters mentioned in the above section are calibrated. QP optimization method
of 50 steps with the iteration step length 0.02 is used in the iteration. All the updating parameters iteration results as is shown in Figure 4.2 and frequency residual error shown in Figure 4.3.

**Figure 4.2** Iteration of all the parameters

**Figure 4.3** Iteration of frequency residual error

The Figure 4.2 shows that each parameter convergences after about 50 steps iteration. Figure 4.3 shows that frequency residual error keep still after 50 steps iteration. The 6th order modal frequency residual error is the largest, the maximum value is about 11%. Each order MAC value basically keep still during the iteration process. The initial finite element model of the overall bogie has high accuracy. The absolute average of frequency error is 9.13% and 3.68%, before and after updating, respectively. The selection of these 6 parameters in the model updating is reasonable, and convergent results are obtained after updating.

A FEM of bogies system is finally established based on validated component structures, i.e. updated frame, wheel set and stiffness parameters of the suspension components. Modal testing of the assembled bogie system with simulated car-body is then conducted on an electrohydraulic shaking table. First six modal frequencies, of 3 rigid body modes and 3 flexible modes respectively, estimated from modal testing are compared with their counterpart computed from bogie FEM. 11 stiffness parameters of the suspension components are selected as parameters to be updated. Model updating is conducted again for the combined bogie system. After model updating, the errors of modal frequencies of all three rigid-body modes are deduced significantly, as shown in the following table:

<table>
<thead>
<tr>
<th>Mode No.</th>
<th>EMA (Hz)</th>
<th>FEM, before updating (Hz)</th>
<th>Freq. errors (%) before</th>
<th>FEM after updating (Hz)</th>
<th>Freq.errors (%) after</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>11.55</td>
<td>10.97</td>
<td>-6.08</td>
<td>11.59</td>
<td>0.37</td>
</tr>
<tr>
<td>2</td>
<td>13.25</td>
<td>13.36</td>
<td>0.98</td>
<td>13.25</td>
<td>0.04</td>
</tr>
<tr>
<td>3</td>
<td>18.09</td>
<td>16.34</td>
<td>-10.88</td>
<td>17.36</td>
<td>-4.48</td>
</tr>
</tbody>
</table>

The errors of the modal frequencies, especially the first three rigid body modes, have been significantly reduced after model updating. The maximum frequency error is reduced from above 10% to less than 5%!

5. CHANGES OF FRAME STRUCTURE MODAL IN THE FREE AND THE OVERALL CONDITIONS

Using the updated model, the changes of structure mode were obtained through calculation and test of frame structure under different boundary conditions. As is shown in Table 5, compared with the condition of overall, elastic modal frequency of a rail vehicle bogie in the condition of free support has declined due to the change of the restraint stiffness or increase of mass of vibration. First three main test modal frequency of the bogie reduced by about 4.2% and the corresponding frequencies of finite element analysis reduced by about 5.7%. The mode shapes almost keep unchanged. In the subsequent dynamical design of bogie, changes caused by the overall condition should be considered when calculate the bogie frequency.
Table 5 The comparison of bogie elastic modal natural frequency between the free support condition and overall condition

<table>
<thead>
<tr>
<th>Mode No.</th>
<th>TEST Free support (Hz)</th>
<th>TEST Overall (Hz)</th>
<th>Rate of change (%)</th>
<th>Initial MAC</th>
<th>FEA.free support (Hz)</th>
<th>FEA. overall (Hz)</th>
<th>FEA.rate of change (%)</th>
<th>Vibration mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>41.25</td>
<td>40.76</td>
<td>-1.2</td>
<td>0.65</td>
<td>41.86</td>
<td>40.73</td>
<td>-2.8</td>
<td>Frame structure torsion</td>
</tr>
<tr>
<td>2</td>
<td>54.89</td>
<td>51.34</td>
<td>-6.7</td>
<td>0.54</td>
<td>55.65</td>
<td>51.55</td>
<td>-7.6</td>
<td>First vertical bending modal of the horizontal beam with antisymmetric pitching movement of side beam</td>
</tr>
<tr>
<td>3</td>
<td>69.51</td>
<td>66.37</td>
<td>-4.6</td>
<td>0.81</td>
<td>63.3</td>
<td>59.19</td>
<td>-6.7</td>
<td>Second vertical bending modal of the horizontal beam with symmetric pitching movement of side beam</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Average value</td>
<td>Average value</td>
<td>-5.7</td>
<td></td>
</tr>
</tbody>
</table>

6. UPDATING CONCLUDING REMARKS

Concluding remarks are finally given and further work on dynamic model updating and validation of full rail vehicle model will be conducted based on validated bogie and car-body FEMs.

REFERENCES


Methods to obtain reliable car body load assumptions for light rail vehicles using multibody simulations based on measured input

Gero Zechel, Michael Beitelschmidt

Chair of Dynamics and Mechanism Design
Institute of Solid Mechanics
Faculty of Mechanical Science and Engineering
Technische Universität Dresden
Marschnerstraße 30, 01307 Dresden, Germany
gerotzechel@tum.de, michael.beitelschmidt@tum.de

ABSTRACT
Conventional analytic methods to get load assumptions for light rail vehicles, forming the basis for their structural design, have shown large gaps to reality in the past. This paper demonstrates how multibody systems of relevant load cases solved in the time-domain can help to fill these blind spots. Their input can be obtained from and their output has to be verified with measurements. To synchronise way-dependant track and time-dependant vehicle data from different sources, a cross-correlation method is formulated. To pin long-term measurements to the track, a system-theoretical approach is shown, describing the location in the track network as a finite-state machine. Regarding the vehicle model, different ways to model the car body structures are discussed. Because quasi-static loads show a large impact, it is crucial to match the car body stiffness between all of its interfaces, e.g. by applying model order reduction methods on finite element models. Key steps of this process are demonstrated. The paper details how to create close to reality simulation scenarios by using measured tracks as well as measured or synthetic vehicle speed profiles. To cope with the high number of simulation runs, it is shown how the process can be fully automated. Keeping the vehicle measurement, synthesised data, and simulation results in the same reference system and side-by-side in software becomes an advantage in post-processing. It is shown how it can be used to find correlations, dependencies, causes, and effects between all available data. A method to compile relevant load cards is presented.

Keywords: Light Rail Vehicles, Load Assumptions, Track measurement, Vehicle measurement, Model Order Reduction.

1 INTRODUCTION
The development towards higher energy efficiency, transportation capacity, safety, accessibility, and comfort has made modern light rail vehicles (LRV) much more complex than their predecessors. To provide a high ratio of low floor passenger area, equipment of considerable weight was shifted to the roof. The need for portals between cars, large windows, and a maximised number of doors made it difficult to laterally support the resulting inertia loads. At the same time, new bogie concepts with stiffer suspension and less degrees of freedom between bogie and car body increased loads further, as track displacements are more directly applied to the car body structure. The length of the vehicles combined with an often demanding urban topography require well-adjusted articulation systems.

Besides some non-linear components, it is the large angles in these articulations that turn the vehicles into fundamentally non-linear systems and the determination of reliable load assumptions into a major challenge. Multibody systems (MBS) of the relevant load cases require the use of increasingly detailed models of vehicles and tracks, as shown in Figure 1. Their input can be obtained from and their output has to be verified with measurements.
of existing tracks, similar vehicles in service, or prototype vehicles. This paper is showing key excerpts from [1].

2 PRE-PROCESSING OF MEASURED INPUT DATA

While the integration of measured track is well-known [2], the usage of vehicle data like measured speed profiles requires the synchronisation of way-dependant track (T) and time-dependant vehicle (V) data from different sources. If there is time-dependant vehicle sensor data at discrete, equidistant time steps $t_k$ with $k \in \{1, \ldots, m\}$, in particular the vehicle speed $v^V_k$ and the yaw rate $\omega^V_k$ of one bogie, and if there is way-dependant track data at discrete, equidistant steps $s^T_i$ with $i \in \{1, \ldots, n\}$, in particular the horizontal track curvature $\kappa^h_{T_i}$, then the vehicle-side measured track curvature

$$\kappa^h_{V_k} \approx \begin{cases} \omega^V_k v^V_k, & \text{if } v^V_k \geq v^{\text{lim}}, \\ \kappa^h_{V_{k-1}}, & \text{else} \end{cases}$$

(1)

can be calculated at each time step. The travelling distance of the vehicle can be obtained by integration with e.g. the Newton-Cotes formula

$$s^V_k = \sum_{l=1}^k \frac{1}{2f} (v^V_l + v^V_{l-1}) + s^V_0,$$  

(2)

where $f$ is the signals sampling rate. By interpolation of the value pairs $s^V_k$, $t_k$ (with equidistant $t_k$) to get the value pairs $t^*_i$, $s^V^*_i$ (with equidistant $s^V^*_i$), using the linear interpolation

$$t^*_i = t_l + \frac{t_{l+1} - t_l}{s_{l+1} - s_l} (s^V_{i} - s_l) \quad \text{with } s_l \leq s^V_{i} < s_{l+1} \text{ for all } i \in \{1, \ldots, n\},$$

(3)

a way-dependant vehicle-side measured track curvature

$$\kappa^h_{V^*_i} = \kappa^h_{V_k} \circ t^*_i.$$  

(4)

can be obtained. To finally get the offset between the vehicle-side and track-side data, the maximum of the cross-correlation function $(\kappa^h_{T_i} \ast \kappa^h_{V^*_i})_l$ of both way-dependant track curvatures,

$$s^T_k = s^V_k + \Delta s_{VT}, \text{ with } \Delta s_{VT} = \arg \max_l (\kappa^h_{T_i} \ast \kappa^h_{V^*_i})_l : \Delta s,$$  

(5)

has to be determined. As an example, Figure 2 shows three measurement runs of a vehicle on the same track segment with different driving speeds, that have been aligned to a independently measured track curvature.

To pin long-term measurements like [3] precisely to the track, an extended approach is needed, because it is unknown to which track sequence the vehicle measurement has to be
LRVs travel on isolated track networks with limited numbers of track segments and switches. Because it is known that (a) the vehicle’s first wheel-set is always located at exactly one track segment of the network and that (b) the vehicle stays on a track segment or changes to a connected track segment between two time steps, the vehicle location can be described as a finite-state machine, with track segments forming states and switches forming transitions.

Taking different directions on a single track into account by doubling the amount of states, the resulting finite-state machine is shown in Figure 3. Because GPS signal loss often veils the first and the last position of a vehicle, an algorithm has been created that obtains the most probable track sequence by global optimisation, regarding all possible track sequences (now state sequences).

Let $\mathcal{S} = \{1, \ldots, n\}$ be the state space and $\mathbf{T} \in \{0,1\}^{n \times n}$ be the matrix that describes all allowed transitions between states, and let $\mathbf{P}$ be a matrix of penalty values $p_{k,i}$ for $k \in \{1, \ldots, m\}$ time steps and $i \in \mathcal{S}$ states. As penalty values, the distance between each GPS position, measured at time $t_k$ in Cartesian coordinates $x_k^{\text{GPS}}, y_k^{\text{GPS}}$, and each state $i \in \mathcal{S}$, described by $l \in \{1, \ldots, o_i\}$ sampling points with coordinates $x_{i,l}^{\text{GPS}}, y_{i,l}^{\text{GPS}}$,

$$p_{k,i} := \min_{l \in \{1, \ldots, o_i\}} \sqrt{(x_k^{\text{GPS}} - x_{i,l}^{\text{GPS}})^2 + (y_k^{\text{GPS}} - y_{i,l}^{\text{GPS}})^2}$$

for all $k \in \{1, \ldots, m\}$, $i \in \{1, \ldots, n\}$, (6)

can be used.

As every state $i$ at time step $t_k$ with $k > 1$ can only be reached if certain (also penalised) preceding states have been reached, a cumulative penalty matrix can be build line-by-line. Would there be no transitions between states, these penalty values would be calculated as

$$p_{k,i} = p_{k,i} + p_{k-1,i}$$

for all $k \in \{2, \ldots, m\}$ (7)
because the vehicle would have to remain in every assumed state till the end. Now, because there are transitions, it happens that a preceding state of \( i \) different to \( i \) has a lower penalty value at \( t_{k-1} \) that \( i \) itself. In these cases, it can be said without limiting the probability of any other track sequence, that the vehicle has most probably reached the state by changing from the preceding state with the lowest penalty value. \( T \) describes all states \( J_i \subset S \) that come into question. These sets \( J_i \) are constant and do also contain \( i \). Therefore, the cumulative penalty values can be calculated for all \( k \in \{2, \ldots, m\} \) using

\[
p_{k,i} = p_{k,i} + \min_{j \in J} p_{k-1,j} \quad \text{for all } i \in S
\]  

instead of Equation 7. The preceding states

\[
W_{k,i} = \arg \min_{j \in J} p_{k-1,j}
\]  

can be collected in \( W \in \mathbb{Z}^{m \times n} \). The state \( i \) that shows the lowest cumulative penalty value at the last time step \( t_m \) is the most probable end state of the measurement run, therefore the last element of the state (or track) sequence \( R \) is

\[
R_m = \arg \min_{i \in S} W_{m,i}.
\]

Running backwards through \( W_{k,i} \) from \( m \) to 2, the whole sequence

\[
R_{k-1} = W_{k,R_k} \quad \text{for } k \in \{2, \ldots, m\}
\]

can be read out. Figure 4 shows an example result. The sequence can be used to compile the horizontal track curvature \( \kappa^H_T \) of the measurement run and apply the aforementioned cross-correlation method. As even small errors in speed information lead to large distance
drifts for day-long measurements, the offset is not constant, but dependant on time (or distance). For its determination, a recursive correlation algorithm has been developed that follows the divide-and-conquer principle. An example of the resulting offset for a long track is shown in Figure 5.

3 VEHICLE MODELING

Regarding the vehicle model, accurate car body modelling is crucial to get precise car body load assumptions. Because quasi-static loads like low-frequency twisting show a large impact, it is important to match the car body stiffness between all of its interfaces. This is often achieved by splitting the rigid body in at least two parts connected with a equivalent stiffness. While this approach shows reasonable results in some load cases for some vehicle concepts, it is strongly recommended to use elastic bodies based on finite element (FE) models instead, as shown in Figure 6. This can be done by applying model order reduction methods to each of the car body FE models, which can be described as second order ordinary differential equations

\[ M \ddot{z} + Kz = F \]  

or, with linear damping, as

\[ M \ddot{z} + D \dot{z} + Kz = F. \]  

For this, it is necessary to split the Degrees of Freedom (DoF) of all nodes of the full FE model in master DoF that have to remain in the reduced model and slave DoF that are described dependently. At least, the DoF of all nodes where external forces are applied on the single car body (sometimes called the car body’s interface forces) have to be defined as master DoF. Further, a transformation matrix \( T \) is needed that sets the master and slave
DoF of the full model in relation to the master DoF only:

$$\mathbf{z} = \begin{pmatrix} z_m \\ z_s \end{pmatrix} = \mathbf{T} \cdot \mathbf{z}_m.$$  \hfill (14)  

Finding a suitable transformation matrix is the core task of model order reduction. In general, the equation system of the full model has to be sorted by master and slave DoF. Without changing the undamped model, the equation system becomes

$$
\begin{pmatrix}
M_{mm} & M_{ms} \\
M_{sm} & M_{ss}
\end{pmatrix}
\begin{pmatrix}
\ddot{z}_m \\
\ddot{z}_s
\end{pmatrix} + 
\begin{pmatrix}
K_{mm} & K_{ms} \\
K_{sm} & K_{ss}
\end{pmatrix}
\begin{pmatrix}
z_m \\
z_s
\end{pmatrix} = 
\begin{pmatrix}
F_m \\
F_s
\end{pmatrix}.
\hfill (15)
$$

with $F_s = 0$, because external forces on the internal nodes are not allowed by definition.
For the static case the transformation matrix can be determined by means of static condensation. Because in this case there is $\ddot{\mathbf{z}} = 0$, the inertia term can be dropped. Therefore,

$$(K_{sm} \quad K_{ss}) \begin{pmatrix} z_m \\ z_s \end{pmatrix} = \mathbf{F}_s = 0 \hfill (16)$$

can be extracted from Equation 15, rearranged by $\mathbf{z}_s$ and inserted into Equation 14, leading to the static transformation matrix

$$
\mathbf{T}_{\text{stat}} = \begin{pmatrix}
\mathbf{I} \\
-K_{ss}^{-1} \cdot K_{sm}
\end{pmatrix}. \hfill (17)
$$

Now $\mathbf{K}$ of the full model can be transformed to get the stiffness matrix

$$
\mathbf{K}_{\text{red}} = \mathbf{T}_{\text{stat}}^T \cdot \mathbf{K} \cdot \mathbf{T}_{\text{stat}} \hfill (18)
$$

of the reduced model, leading to the reduced, static model

$$
\mathbf{K}_{\text{red}} \mathbf{z}_m = \mathbf{F}_m \hfill (19)
$$

which precisely covers the static properties of the full model when forces are applied on the master DoF.

Reduction by the Guyan-Method is based on the assumption, that the mass matrix $\mathbf{M}_{\text{red}}$ and, if applicable, the damping matrix $\mathbf{D}_{\text{red}}$ can be computed analogical to Equation 18 using

$$
\begin{align*}
\mathbf{M}_{\text{red}} &= \mathbf{T}_{\text{stat}}^T \cdot \mathbf{M} \cdot \mathbf{T}_{\text{stat}} \quad \text{and} \\
\mathbf{D}_{\text{red}} &= \mathbf{T}_{\text{stat}}^T \cdot \mathbf{D} \cdot \mathbf{T}_{\text{stat}} \hfill (20)
\end{align*}
$$
to get to a reduced, dynamic model

\[ \mathbf{M}_{\text{red}} \ddot{\mathbf{z}}_m + \mathbf{D}_{\text{red}} \dot{\mathbf{z}}_m + \mathbf{K}_{\text{red}} \mathbf{z}_m = \mathbf{F}_m \]  

(21)

that covers the full model at least for low frequencies.

To cover the dynamic behaviour of the full model over a high range of frequencies, there’s a plurality of dynamic reduction methods, usually based on static condensation, like Component Mode Synthesis (CMS). For further reading see [4] and [5]. When using dynamic reduction methods, it has be kept in mind that an improved dynamic representation comes by the cost of the static accuracy, so the quality of results can be reduced for the highly relevant static and quasi-static load cases.

4 CREATING SIMULATION SCENARIOS

Close to reality simulation scenarios like displayed in Figure 7 can be created by using measured tracks combined with synchronised, measured vehicle speed profiles. If measured track data is available without measured speed profiles, synthesised speed profiles can be used instead. The vehicle speed is limited by the maximum allowed lateral acceleration \(a_{q,\text{max}}\) in curves (taking into account cant of super-elevated rails), the maximum allowed speed \(v_{\text{max}}\), the longitudinal limits for acceleration and breaking (\(a_{x,\text{max}} > 0\) and \(a_{x,\text{min}} < 0\)). Is track curvature and cant available at way-dependant equidistant steps \(s_i\) with \(\Delta s = s_i - s_{i-1}\) for each \(i \in \{2, \ldots, n\}\), the following approach to get the speed profile can be used.

\[ a_{q,i} = v_i^2 \cdot \kappa_i - \frac{u_i}{b} g, \]  

(22)

For taking positive cant into account without lowering the speed in case of unwanted, negative cant, for all \(i \in \{1, \ldots, n\}\)

\[
v_i := \begin{cases} \sqrt{\frac{1}{\kappa_i} a_{q,\text{max}} + \frac{1}{\kappa_i} \frac{u_i}{b} g}, & \text{if } \text{sgn}(\kappa_i) = \text{sgn}(u_i), \\ \sqrt{\frac{1}{\kappa_i} a_{q,\text{max}}}, & \text{else} \end{cases}
\]  

(23)

can be applied. To now obey maximum speed,

\[
v_i := \min(\{v_i, v_{\text{max}}\}) \quad \text{for all } i \in \{1, \ldots, n\}
\]  

(24)

![Figure 7. Example load case of a vehicle on measured track of around 6 km](image)

The lateral acceleration \(a_q\) at track position \(s_i\) depends on the vehicle speed \(v_i\), the horizontal curvature \(\kappa_i\), the track cant \(u_i\) and the track gauge \(b\) according to
Figure 8. Graphical output of the speed synthesis routine

can be used. To limit the current positive acceleration between to steps \(i\) and \(i + 1\) of

\[
a_{x,i,i+1} = \frac{v_{i+1}^2 - v_i^2}{2\Delta s}
\]  \quad (25)

to the maximum allowed acceleration \(a_{x,max}\), \(v_{i+1}\) has to be lowered to

\[
v_{i+1} := \begin{cases} 
\sqrt{v_i^2 + 2a_{x,max}\Delta s}, & \text{if } a_{x,i,i+1} > a_{x,max}, \\
v_{i+1}, & \text{else}, 
\end{cases}
\]  \quad (26)

if necessary. In the same way, for negative acceleration,

\[
v_{i-1} := \begin{cases} 
\sqrt{v_i^2 - 2a_{x,min}\Delta s}, & \text{if } a_{x,i-1,i} < a_{x,min}, \\
v_{i-1}, & \text{else}
\end{cases}
\]  \quad (27)

has to be applied for all \(i \in \{2, \ldots, n\}\) in descending order. Because the speed profile has to be considered at all wheel-sets, the vehicle length \(l_{veh}\) between the first and the last wheel-set, measured as \(l \in \mathbb{Z}\) steps with a length of \(\Delta s\), is used to compute

\[
v_i := \min_{j \in \{i-l, \ldots, i\}} v_j \quad \text{for all } i \in \{l+1, \ldots, n\}. \quad (28)
\]

Figure 8 gives an example for the resulting speed profile. The according non-equidistant time steps

\[
t_i := \begin{cases} 
0 - \frac{l_{veh}}{v_a} & \text{for } i = 1, \\
t_{i-1} + \frac{2\Delta s}{v_{i-1} + v_i} & \text{for all } i \in \{2, \ldots, n\}
\end{cases}
\]  \quad (29)

are taking into account that at the beginning of the simulation \(t = 0\), the vehicle is already at position \(s = l_{veh}\). Using linear interpolation, the time-dependent speed input

\[
v_k^* = v_j + \frac{v_{j+1} - v_j}{t_{j+1} - t_j}(t_k^* - t_j) \quad \text{with } t_j \leq t_k^* < t_{j+1} \text{ for all } k \in \{1, \ldots, m\} \quad (30)
\]
and the time-dependant distance input

\[ s_k^* = s_j + \frac{\Delta s}{t_{j+1} - t_j} (t_k^* - t_j) \quad \text{with } t_j \leq t_k^* < t_{j+1} \text{ for all } k \in \{1, \ldots, m\} \]

(31)
can be obtained at equidistant \( t_k^* = (k - 1) \cdot \Delta t^* \) and can be set for the simulation.

As the non-linearity rules out the superposition of load cases, \( O^n \) load cases have to be taken into account to cover \( n \) factors of influence. To cope with this high number of simulation runs, it is inevitable to automate the process. Although there is a wide range of entities that can and have to be varied, it is possible to reduce them to a 2D-simulation-matrix of model variants and load cases in pre-processing. This allows for a straightforward batch processing routine, as summarised in Figure 9.

5 EVALUATION METHODS

Following the thesis that the most relevant car body load cases for light rail vehicles are of a static or quasi-static nature, it is possible to get to time-dependant strain curves by stringing together strain calculations at single time steps on the non-reduced model, using approaches like inertia relief [6] to cover quasi-static inertia forces. Figure 10 shows a plot of car body strain that has been computed this way using a step width of 0.5s, alongside strains that were measured in reality by strain gauges at identical positions.

Alongside to measured and simulated strains or stresses, all other data that is stored in the same reference system can be visualised, as shown in Figure 11. It displays stress in the car body structure (first plot) along the difference of spring deflection of the left and right secondary suspension between bogies and car bodies (second plot). These deflections
correspond to the torsional loads on the car body, often induced by large track excitations. As vehicle and track data is synchronised, track radii and cant can be shown at each bogie (third plot). Thus, the example gives a complete picture of strain and loads (effects) as well as the vehicle state and the track excitation (causes).

To further formalise the evaluation of simulated load cases to get to load assumptions for car body design, let \( F_{i,x}, F_{i,y}, F_{i,z} \) be all forces at all \( n \) car body interface nodes (joints, suspension, damper consoles etc.) and \( M_{i,x}, M_{i,y}, M_{i,z} \) the according torques, and let \( r_{i,x}, r_{i,y}, r_{i,z} \) be the location of these interface nodes in a body fixed coordinate system, with \( i \in \{1, \ldots, n\} \). Then these interface loads form a \( 6n \)-element load vector on the car body depending on time. This load vector usually doesn’t contain the max. values of all loads at the same time, and not all time steps can be examined for structural design. For fatigue calculation against the endurance limit, a stress amplitude, consisting of a lower and upper stress value, has to be determined at each point of the car body structure, especially at welding seams and other notches. Therefore, it is important to find a reduced number of load vectors that lead to high overall car body load. This calls for appropriate assessment values.

First of all, by summarising the interface loads component-wise, global forces and torques on the car body can be gained, that are compensated by inertia loads on the car body. Thus, these global loads are directly related to longitudinal acceleration, lateral acceleration, e.g. in curves, vertical acceleration (including gravity), e.g. by track excitation and angular accelerations for pitching, rolling, and, most importantly, yawing (e.g. at curve entry). They can be computed at all time steps \( t_k \) and stored as the first assessment loads \( E_{1,1,k} \) to \( E_{1,6,k} \).

Additionally, as stated e.g. in [7], load states are important that lead to high torsion, shearing and bending of the car body as a whole. To get the according time steps in good approximation, a general approach has been developed that splits the car body in two halves along all three dimensions. This splits all car body loads in two groups

\[
A_x = \{i \in \{1, \ldots, n\} \mid r_{i,x} \leq 0\} \quad \text{and} \\
B_x = \{i \in \{1, \ldots, n\} \mid r_{i,x} > 0\},
\]

for splitting the \( x \)-axis at the \( yz \)-layer, sets \( A_y, B_y \) for splitting at the \( xz \)-layer, and sets \( A_z, B_z \) for splitting at the \( xy \)-layer.

At each of these layers, 6 interface loads can be assumed, that, again at the \( yz \)-layer, form...
Figure 11. Example output of synchronised car body stresses, spring deflections and track excitation, showing effects and causes the assessment values

\[ E_{2,1,k} = \frac{1}{2} \left( \sum_{i \in A_x} F_{x,i,k} - \sum_{j \in B_x} F_{x,j,k} \right), \quad (34) \]
\[ E_{2,2,k} = \frac{1}{2} \left( \sum_{i \in A_x} F_{y,i,k} - \sum_{j \in B_x} F_{y,j,k} \right), \quad (35) \]
\[ E_{2,3,k} = \frac{1}{2} \left( \sum_{i \in A_x} F_{z,i,k} - \sum_{j \in B_x} F_{z,j,k} \right), \quad (36) \]
\[ E_{2,4,k} = \frac{1}{2} \left( \sum_{i \in A_x} (M_{x,i,k} + r_{y,i} F_{z,i,k} - r_{z,i} F_{y,i,k}) - \sum_{j \in B_x} (M_{x,j,k} + r_{y,j} F_{z,j,k} - r_{z,j} F_{y,j,k}) \right), \quad (37) \]
\[ E_{2,5,k} = \frac{1}{2} \left( \sum_{i \in A_x} (M_{y,i,k} + r_{z,i} F_{x,i,k} - r_{x,i} F_{z,i,k}) - \sum_{j \in B_x} (M_{y,j,k} + r_{z,j} F_{x,j,k} - r_{x,j} F_{z,j,k}) \right), \quad (38) \]
\[ E_{2,6,k} = \frac{1}{2} \left( \sum_{i \in A_x} (M_{z,i,k} + r_{x,i} F_{y,i,k} - r_{y,i} F_{x,i,k}) - \sum_{j \in B_x} (M_{z,j,k} + r_{x,j} F_{y,j,k} - r_{y,j} F_{x,j,k}) \right). \quad (39) \]

When splitting at the xy-layer and the xz-layer, \( E_{3,1,k}, \ldots, E_{3,6,k} \) and \( E_{4,1,k}, \ldots, E_{4,6,k} \) can be computed. With the resulting 24 assessment loads for \( t_k \) there are 48 time steps \( t_K \) with

\[ K := \arg \min_{k \in \{1, \ldots, m\}} E_{g,i,k} \quad \cup \quad \arg \max_{k \in \{1, \ldots, m\}} E_{g,i,k} \quad \text{for} \quad g \in \{1, \ldots, 4\}, \quad i \in \{1, \ldots, 6\}, \quad (40) \]
where the minimum or maximum of each assessment load is reached and a high overall car body load can be assumed. These 48 load vectors can be compiled in load tables, describing the load assumptions and thus forming the input for detailed car body structure design.

6 CONCLUSIONS

By complementing measured tracks with measured or synthesised vehicle speed profiles from short or long-term vehicle measurements, close to reality simulation scenarios can be built for multibody simulation. It can be shown that the underlying synchronisation of track and vehicle data can be achieved with a cross-correlation method for short measurements and that this can be extended for long-term measurements with a system theoretic approach.

A highly automated process has been set up for pre-processing, simulation, and evaluation to cope with a high number of load cases, which is needed because non-linearities forbid the superposition of results of just a few basic load cases. By building a vehicle model that covers the car body stiffness at least for low frequencies by applying static model order reduction on finite element car body models, structural strain on critical sections has been predicted with good accuracy.

The combined evaluation of measured and simulated data like car body strain, vehicle states and track information leads to the ability to recognise causes for high loads and thus makes it possible to choose optimal vehicle set-ups to reduce them. The reliable computation of car body load assumptions for individual vehicle concepts and track networks continues to be a key aspect in light rail vehicle design to succeed in building lighter, larger, and more efficient vehicles for urban transport.

REFERENCES


Modal Testing and Parameter Identification of an Overall Rail-vehicle Carbody of a High-speed Train for FE Model Validation

Qintao Guo*, Dandan Yang*, Haitao Li#, Yanju Zhao*, Tong Wang*, Baoqiang Zhang*, Lingmi Zhang*

* Department of College of mechanical and electrical engineering, Nanjing University of Aeronautics & Astronautics
29 Street Yudao, 210016, Nanjing, China
guo_qintao@nuaa.edu.cn

#CSR Qingdao Sifang Locomotive & Rolling Stock Co., Ltd.
No.88 in Jinhong east road of Chengyang district
266000, Qingdao, China
lihaitao@cqsfl.com

ABSTRACT
Duo to the varieties of stiffness and damping elements in the overall rail-vehicle system, the performance of those parameters have significant effect on NVH. Experience shows that the stiffness difference in the primary mental spring is more than 3% and the secondary air spring reaches more than 5%. The stiffness and damping of some connecting components also show nonlinearities in frequency domain because the vehicle speed variation and quality changes in operating state. In order to verify the overall rail-vehicle system, advanced experimental modal analysis technique based on multiple instrumented excitations is utilized. Modal parameters identified from EMA are validated via various measures, such as reciprocity check via cross FRFs, FRF curve fittings, etc. Based on the model validation and multistage calibration, this article focused on the model validation and calculation of nonlinear changes and uncertainty distribution of important connecting components’ vibration response, by building and validate the vehicle’s FE model.

Keywords: vehicle dynamics, air spring, frequency-dependent nonlinear, finite element model validation, uncertainty

1 INTRODUCTION
The precision and reliability of the overall rail-vehicle system dynamics model of a high-speed train accounts significantly as the prerequisite for the calculation of vibration and comfort. Varieties of stiffness and damping elements, such as the primary mental spring, the secondary air spring, rubber joint, etc., the performance of which have significant effect on NVH. Due to the influence of manufacturing and assembling errors, obvious uncertainties lay in the stiffness and damping properties of connecting components. Experience shows that the stiffness difference in the primary mental spring is more than 3% and the secondary air spring reaches more than 5%. Besides, the stiffness and damping of some connecting components also show nonlinearities in frequency domain because the vehicle speed variation and quality changes in operating state. Based on the model validation and multistage calibration, this article focused on the model validation and calculation of nonlinear change and uncertainty distribution of important connecting components’ vibration response (such as air spring), by building the vehicle’s FE model.

2 BASIC THEORY
2.1 RSM Based Calibration for Selective Parameters
Different to correction and calibration method based on sensitivity analysis, the response surface methodology (RSM) involve in computing response features of every design point in the space spanned by the parameters according to DOEs (Design of Experiments) [2]. Then a high order polynomial model (or other RS model) is regressed; There are 5 main steps: ① In design space, the response can be calculated by the finite element analysis according to the design of experiments(such as Latin square, D-optimal design); ② Use computational modal analysis to generate the response features(such as modal frequency) ③ RS models can be constructed by the Least Square method or other moden metamodel method, and the RS model verification should be conducted to decide whether the model has enough accuracy or not. ④ RS based Sensitivity analysis was employed, or the optimization problem can be solved by traditional optimization method and the response
feature of the structure should be measured which can be taken as the aim feature. Iteration of model updating can be conducted within the RS model, and Calibration parameters can be obtained and transferred to the FE model.

One more thing needs to be mentioned before RS regression we can choose other features such as maximum deformation or maximum accelerations as the additional response features to build more generalized RS models.

The obtained RS model can be expressed as

\[ y_j(p) = f_j(p) + \varepsilon_j \]  

(1)

The optimization problem can be expressed as

\[
\begin{align*}
\min_{p} & \| R(p) \| \\
\text{s.t} & \quad p \leq p \leq p_a
\end{align*}
\]

(2)

Where \( f_j(p) \) — RS models

\( \varepsilon_j \) — Regression error

\( p \) — Parameters that needs to be updated.

\( \{y_x\}, \{y_a\} \) — Experimental features and analytical features

\( R \) — Residue,

\( p_l, p_a \) — Bounds of the parameters.

### 2.2 Distribution Estimation of the Response Features Prediction

The direct solution to compute the Distribution of response features is LHS (Latin hypercube sample) \(^{[3,4]}\) integrated with RS models. The parameters were treated as norm distribution and intervals \(^{[1, 5]}\). LHS is a sampling method that generates sample points randomly and hierarchically. Given m samples points of n variables is needed, the sampling progress is described as following. Divide every variables with m equal intervals which incises the design space into \( n \times m \) subspaces and then randomly sampling is conducted in these subspaces. The method used in hierarchy is based on equal probability size which is much simpler than that used in stratified sampling. MC is a random sampling campaign and the sample points are randomly distributed in the design space. Gathering occurs when the sample size is relatively small in MC. But this phenomenon is avoided in samples utilizing LHS which stratifies before sampling. The LHS X contains n values on each of p variables. For each column, the n values are randomly distributed with one from each interval \((0,1/n), (1/n,2/n), \ldots, (1-1/n,1)\), and they are randomly permuted. Compared with MC sample which will be not random due to the sample size being smaller, the LHS need divide the space of the parameters into equal probabil subspaces, then do the MC sampling. After the LHS sampling, the distribution of the Response Features was estimated by the normal fit method or the other distribution fit method.

### 2.3 Model validation criteria of the experiment and calculation distribution

The validation comparisons, in this case, involve comparison of the single output from the system model to the distribution of experimental outputs. The validation metric used here is to estimate the difference between an experimental response and the estimated mean of the population of the response by the computational model. That is a sample and a single response. Using the traditional level of confidence of 95%, one can state the validation metric in the following way.\(^{[6]}\)

\[ x \in [\mu(y) - 1.96\sigma_y, \mu(y) + 1.96\sigma_y] \]

When the sample is small, the following Eq.(2) can be used\(^{[6]}\) :
In which $x$ stands for the experimental response feature, $y$ stands for the computational response. $\mu$ is the estimate mean of the population of the response by the computational model. $\sigma_y$ is the estimate standard variance of the population.

3 EXPERIMENTAL CASE STUDY

3.1 Identification and Calibration of the Key Parameters.

According to the process of model validation, we obtained accurate parameter distribution of stiffness and damping during the study of each air spring and mental spring. The typical nonlinear characteristics of air spring is shown in Fig.1, which is considered in the establishment of nonlinear FE model of the vehicle, and the stiffness distribution characteristics of mental spring is shown in Fig.2. Both mental spring and air spring uncertainties are obtained through individual test, then the probability distribution was fitted by probability distribution testing. Besides, the distribution characteristics of the middle mass of the car body is shown in Fig.3 and the overall sampling 8 points of three parameters are shown in Fig.4.
3.2 Experimental Modal Analysis of the Whole Vehicle.

Vehicle modal experiment is a critical challenge work which requires proper excitation and acceleration-testing distribution of measured points, as well as advanced EMA identification method and software to process the data. We excited multi-points on the vehicle at the same time to conduct modal test, and the main modal are identified in 0-40Hz. A typical FRF of EMA is shown in Fig.5. 6 modal parameters were extracted from the experimental model which can be used in this validation. The 6 modal frequencies shown in Table 1.

![Fig. 5 A typical FRF of EMA of the vehicle in overall conditions](image)

**Table 1** The six experimental and FEA mode frequencies(after model updating)

<table>
<thead>
<tr>
<th>Mode No.</th>
<th>EMA (Hz)</th>
<th>FEM with linear stiffness (Hz)</th>
<th>Name of the shape</th>
<th>Freq.errors (%) after updating</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.94</td>
<td>0.92</td>
<td>rolling movement</td>
<td>-2.1%</td>
</tr>
<tr>
<td>2</td>
<td>1.48</td>
<td>1.40</td>
<td>bouncing movement</td>
<td>-5.4%</td>
</tr>
<tr>
<td>3</td>
<td>1.60</td>
<td>1.75</td>
<td>pitching movement</td>
<td>9.3%</td>
</tr>
<tr>
<td>4</td>
<td>12.77</td>
<td>12.83</td>
<td>distortion</td>
<td>0.4%</td>
</tr>
<tr>
<td>5</td>
<td>13.30</td>
<td>13.33</td>
<td>bending</td>
<td>0.2%</td>
</tr>
<tr>
<td>6</td>
<td>14.40</td>
<td>14.48</td>
<td>torsion</td>
<td>0.6%</td>
</tr>
</tbody>
</table>

3.3 Validation of the FE Model of Overall Rail-Vehicle Car-Body with EMA Results.

During the simulation, air springs were set as frequency-dependent nonlinear and other connecting members were set as linear uncertainty parameters. The FE model of the overall rail-vehicle car body including these connecting is shown in Fig.6. Then the modal parameters calculation is conducted by obtained distribution of their characteristics. The distribution is given through the calculation of Monte Carlo simulation by LHS sample. The 8 sampling stiffness nonlinear FRFs and a linear FRF were shown in Fig.7. Finally, compared with the modal experiment results (shown in Table 2), the FE model of vehicle satisfies the verification conditions within the specific frequency band, mode 2,3 is out of the region. The first three rigid modes were not sensitive to the selected 3 parameters.
![Fig.6 The FE model of vehicle in overall conditions](image)

**Table 2** Comparison of the six experimental frequency and FEA frequencies interval

<table>
<thead>
<tr>
<th>Mode No.</th>
<th>EMA (Hz)</th>
<th>FEM intervail (Hz)</th>
<th>In or out</th>
<th>Mode No.</th>
<th>EMA (Hz)</th>
<th>FEM intervail (Hz)</th>
<th>In or out</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.94</td>
<td>[0.85 0.95]</td>
<td>In</td>
<td>4</td>
<td>12.77</td>
<td>[12.19 13.23]</td>
<td>In</td>
</tr>
<tr>
<td>2</td>
<td>1.48</td>
<td>[1.35,1.45]</td>
<td>out</td>
<td>5</td>
<td>13.30</td>
<td>[12.94 13.63]</td>
<td>In</td>
</tr>
<tr>
<td>3</td>
<td>1.60</td>
<td>[1.65,1.75]</td>
<td>out</td>
<td>6</td>
<td>14.40</td>
<td>[13.92 14.88]</td>
<td>In</td>
</tr>
</tbody>
</table>

**Fig. 7** The 8 sampling stiffness nonlinear FRFs of FE model for the vehicle in overall conditions

**3.4 Response Prediction of the Vehicle under the Typical Excitation Spectrum.**

Put forward the FEM response prediction of the vehicle under the typical excitation spectrum (vibrating test simulated road-spectrum) the FEA response spectrum was shown in Fig.8. The prediction of the vehicle response spectrum RMS value was obtained by the random spectrum analysis as shown in Fig.9. RMS changes due to the parameters’ differences, as shown in Table 3.
Fig. 8 Vibrating test simulated road-spectrum of rail-vehicle

Fig. 9 The power spectral density response of the FE model

<table>
<thead>
<tr>
<th>Table 3</th>
<th>RMS value in different parameter conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RMS (g)</td>
</tr>
<tr>
<td>0-50Hz</td>
<td>6.112E-5</td>
</tr>
</tbody>
</table>

4 CONCLUSIONS AND REMARKS

The validation method of FEM we used is advanced in dealing with the uncertain parameters and the nonlinear response analysis effectively. It was implied that the stiffness of the connection and the mass difference have little influence to the rigid mode. But they have more influence to the lower structural modes. Through the statistical analysis, the RMS random spectrum change was given due to the parameters’ differences. We will evaluate and optimize the vehicle vibration and comfort through the optimal design and robust design in the future.

ACKNOWLEDGEMENT

This work is supported by the Chinese National Natural Science Fund: No.51275240.

REFERENCES


Network-centric approach to adaptive real-time train scheduling

Alexander A. Belousov*, Petr O. Skobelev*, Maxim E. Stepanov*

* Samara State Aerospace University (National Research University)
34, Moskovskoye shosse, Samara, 443086, Russia
belousov@smartsolutions-123.ru

# SEC “Smart Solutions”
443013, 17 Moscovskoe Shosse, office 1201, Samara, Russia

ABSTRACT
The paper proposes a network-centric approach to creating intelligent adaptive system of real-time train scheduling on the basis of multi-agent technologies. The architecture of the network-centric multi-agent system consisting of base planning subsystems is described. Subsystem interaction protocols and protocols of agent interaction within each subsystem are presented. The example of schedule planning in various situations is presented. Productive characteristics of the developed system are presented. Good quality of schedule planning and system performance is shown.

Keywords: railway dispatcher systems, train scheduling, multi-agent systems, network-centric approach, intelligent systems, method of conjugate interactions, adaptive planning.

1 INTRODUCTION
When we solve complex automation tasks of railway real-time train scheduling, we constantly have to deal with a lot of disruptive events, which differ in intensity and source and can lead to conflicts in schedules of different kinds of trains. This makes matters even worse in large-scale railway systems with a high level of connectivity. Today solving conflict situations in trains deviating from master schedule completely depends on the experience of the train dispatcher, which often leads to irrational decisions, especially in stressful situations. The constantly growing intensity and speed of the passenger traffic and hence results in increasing complexity of disruptive situations. This also provokes questions as how to reduce dependency on the human factor by automating the decision-making process, how to build intelligent system, enabling fast and effective adjustments in case of a disruptive event.

Train re-scheduling during railway traffic disturbances is an important problem. A Mixed-Integer Linear Program (MILP) model is proposed for train re-scheduling of N-tracked railway traffic during disturbances in [1]. In [2], the same model as proposed in [1] is used along with two solution methods: (i) right-shift re-scheduling to produce the initial feasible solution and (ii) local search to limit the search. The train scheduling problem can be formulated as a job shop scheduling problem, as in [3, 4], where train trips are jobs which are scheduled on tracks that are considered as resources. Two studies [3, 5] addressed the problem from the perspectives of capacity, robustness, and dependencies. The heuristics and integer solution methods along with analysis are given in [3]. A variable speed dispatching system is proposed in [4] to control railway traffic by considering acceleration and deceleration time in the model. Furthermore, the work in [4] extends [6] with detailed microscopic and comprehensive models to fulfill additional requirements.

Studies of the computational complexity of disturbance handling in large railway networks are done in [7] and [8]. An experimental study of optimization (i.e. Minimize delay cost) and myopic based policies (i.e. First Come First Served etc.) concludes that both complement each other [7]. The performance of each one is dependent on the region and disturbance type. A performance evaluation of centralized and distributed strategies for dispatching trains is given in [8]. A greedy depth-first search branch-and-bound algorithm is proposed in [9] to handle the re-scheduling problem. It generates a feasible solution within 30 sec in most cases.
This paper describes the network-centric approach to solving the complex task of real-time train scheduling in a large-scale system. Modern large-scale railway train scheduling systems are characterized not only by a large number of participants, but also by high intensity and speed of trains, which is increasing year by year. Due to the high traffic intensity, trains are highly interconnected: changes in one train’s schedule, or a conflict with this train, will most certainly affect the next train, and with high probability have an impact on the whole train network. In this case re-scheduling of all trains in the network might be required, which should be done quickly, on the fly in real-time, which is a very difficult task given the whole variety of planning conditions, preferences and constraints. Constraints must be applied individually and can be regulated in the work process.

The presented network-centric approach was used to develop real-time train scheduling system. This system was adapted for production use in Russian Railways on a section of the high-speed rail Saint-Petersburg - Moscow and between Saint-Petersburg and Buslovskaya.

2 NETWORK-CENTRIC APPROACH TO ADAPTIVE TRAIN SCHEDULING

2.1 Problem statement

The suggested network-centric approach to solving the complex task of train scheduling is aimed at creating optimal schedules for trains, which are delayed due to unforeseen events in real time.

The developed system builds the initial train schedule, and then performs schedule corrections according to different events (infrastructure repair requests, actual train placements, infrastructure elements occupations and so on).

The input data consists of: railway infrastructure (stations, railway switches and block sections of railway infrastructure), requirements for train schedule (master-plan), maintenance requirements, updates on the current situation in the operating domain about trains and states of infrastructure block sections (signals of busy condition, information on unavailability). It is worth mentioning that the scale of the task is enormous which makes it a large-scale task.

The main limitations of the system are traffic security requirements, normative route-building requirements, train priorities, dispatcher rules etc.

Besides the limitations listed above there are ones which are hard to formalize (no thickening of schedule lines of train routes, no unjustified changing of tracks, no traffic jams between stations, no unjustified train stops on the main tracks, correct routing of arrivals etc.), which should be taken into consideration while planning. At the same time, the implementation of any requirements depends on the current situation.

For instance, a train may not choose opposite tracks for movement, but if there is a busy infrastructure block section on its path, then it can do so, in order to bypass the obstacle and stay on. On the other hand, it may stay on the same tracks, but only wait a little, if this delay is not long and the train can catch up its schedule. But waiting at a station means stop for a little while, and a train may only choose block sections of certain length for stay and so on. Hence, a simple decision whether the route should be changed or not, is connected to many conditions, which require finding a balance and consensus. For well-balanced decisions in such situations every train agent in the system is supposed to have collectable virtual “currency” (similar to potential energy), which is used for rewarding successful decisions and, on the other hand, can be spent on fines in case of bad decisions, compensating for the change of the route and the train schedule. This way, all the requirements and limitations which are hard to formalize, can be reduced to a universal measure and be considered in the scheduling of train routes.

The system for solving the task of adaptive real-time train scheduling runs on Vektor-M program platform [10, 11], which allows for keeping the dynamic infrastructure model of the operating domain, get signals from block sections, appointed maintenance windows, satellite and other information.
2.2 Network-centric system architecture

The architecture of the developed system is built on the network-centric principles, where every subsystem has its own individual task and the final solution is reached through negotiating between individual decisions [12].

The primary plan building takes place in 2 subsystems. Each subsystem builds a train route schedule on its own level of understanding of the scene in such way that the initial rough decision is transformed into more precise one. A decision made in every subsystem is conflict-free for its level of understanding (no converging train routes, the security requirements are intact). This layer-based train scheduling eliminates the combinatorial explosion of possibilities, make the scheduling process more stable to disruptions due to reducing the scale of the task on higher levels and step-by-step considering all possible limitations according to the level of importance and impact on other layers.

All events arriving in the system can be divided into two main types: new request or update on the current situation. Requests in their turn can be of the two basic types: request to let the train pass on schedule or request to conduct maintenance works. Update on the current situations can be either a train moving along block sections or a state of an infrastructure (damage or busy condition).

The first level of planning is represented by a trajectory scheduler, the second one by a time scheduler. A general decision-making method and the role of schedulers in it are shown in Figure 1.

The diagram describes interaction between systems on the high level of abstraction. The diagram blocks in their turn can be subsystems with complex inner structure.

Work in the system starts with a message “Start”, which is sent to the agent “Path scheduler agent”. In response the function “Prepare initial schedule” appears, which prepare the planning scene and after which the event “Received conflict” appears in the time scheduler. In the result, the time scheduler detects current schedule conflicts and determines further direction. If no conflicts are found, “Path scheduler agent” is sent a message “All conflicts resolved”, and by processing the message the function “Prepare final schedule” forms the overall schedule. If conflicts are found, there two variants of events: to send the message “Rebuild problem trajectories” to the trajectory scheduler or to solve conflict on its level, by changing time of resources occupation and sending “Resolve conflict” to station and station limits agents.

In the first case, the trajectory scheduler will change tracks and planned block sections of trains’ stops in the schedule problem zone and will send a message “Problem trajectories are re-built” to “Time scheduler agent”; then the agent “Update scene” is called, which start link updating process of requirement-potentials network between requirements of train traffic and infrastructure elements, after which the event “Received conflict” emerges again in the time scheduler.

In the second case, in the process of resolving conflicts in their own schedule, station and station limits agents create conflicts in train schedule, so called “Gap”, which appear due to the difference in arrival and departure time on the adjacent elements of infrastructure. Solving “Gap” conflicts takes place in the method “Close up gap”, implemented by train agents, resulting in new conflicts of resources occupancy, which is checked when event “Received conflict” appears. Reaching a compromise between train agents and agents of infrastructure elements (stations and station limits) is the final objective of the planning system.
If a compromise is reached and all conflicts are resolved, but decision does not satisfy the requirements, a message “Check scene” initialize the process of proactivity, in result of which certain agents try to improve their own schedule and cause a new field of resolving conflicts.

Time scheduler represents an operating domain as a set of station limits and stops, which builds a train schedule in less detail, conflicts are resolved by queueing the trains, speeding them up and slowing them down.

Time planning is similar to a visual schedule analysing. The main task is to build a new possible train schedule considering the normative schedule limitations ant train priorities. The solution is based on the method of conjugate interactions for managing resource allocation in real time [4]. On this level a train agent creates subtasks (operations) for passing a station limit or stay at a station for a certain amount of time [13]. An agent of every subtask of this kind looks for a placement for itself in the respective resource, trying to find the most profitable position by negotiating with other subtask agents. High-priority trains are more active in finding a placement (have more energy for pushing other requests for resources).

The main decision-making condition here is accomplishing the task with minimal divergence (1).

\[ \text{DEVT} = \sum (|TDS - TPS| + |TDF - TPF|) \rightarrow \min, \] (1)
where TDs – scheduled starting time, TDF – scheduled finishing time, TPs – actual starting time, TPf – actual finishing time, N – number of resources (station limits, station platforms), where operations of passing and stopping can be implemented. Additional conditions implemented while making decisions are listed below. The result of the time scheduler system’s work is a schedule for stay and passing stations and station limits in an operating domain, which is sent to the trajectory component in order to build train routes according to infrastructure block sections, provided it is possible.

While scheduling on time scheduler there are two types of interaction:

1) Interaction between interval agents of different trains on one block section: this algorithm enables to provide for an order change of moving trains on block sections;

2) Interaction between interval agents of one train on different block sections. This algorithm enables to keep the trajectory entirety of each train (if a train is late at some block section, it requires correcting the schedule of this train on all its block sections).

Each type of interaction is used for the decision-making of respective type of conflict, and one type of conflict might result in appearance of conflicts of other type.

In the trajectory subsystem a trajectory on block sections is built according to the calculated graphic in less detail, conflicts are resolved due to bypassing and route changing.

The primary task of scheduling paths and stays is to allocate routes for trains to take, and choose the block section for their stays considering overlapping routes of arriving and leaving for the parking. In this subsystem a train agent creates new subtask agents, which look for routes for passing the station and station limit according to the condition of minimal route costs. “Cost” is the cumulative key performance indicator of the route, which includes different normative requirements for train routes (correct or not, length, number of connections). After scheduling with minimal KPIs, station route agents enter the active phase of life cycle, where the main condition for decision-making is no overlapping of train routes in a block section. When such an overlap is found, a station route agent will try to transmit one of the conflicting subtasks to other route agents. Route agents communicate via the task exchange protocol [14, 15].

Building and negotiating the final train schedule takes place in a close interaction between scheduler levels. In every planning subsystem there is a swarm of agents, representing the level, between the subsystems there are back links which come into play when a conflict cannot be resolved locally in the current subsystem. The primary allocation of tasks to resources is done based on the best decision possible independently and in concurrent threads, which allows for reduction of computing time by excluding the rest of possibilities. Such “greedy” allocation results in conflicts that are resolved by agents grouping together into structures within a swarm – domains. In each domain searching for a compromise takes place between agents in order to resolve the conflict.

2.3 Train movement modelling

System builds train schedule and dispatchers’ corrections based on train movement by infrastructure block sections modelling.

Train rout consists of ordered list of infrastructure block sections. These block sections must have connections between them. Besides this spatial continuity, train rout must have time continuity. I.e. infrastructure block section occupation time must be equal to release time of previous block section in a rout. Train rout is built by scheduling system.

Train movement model takes into account train characteristics of acceleration and braking on infrastructure block sections. Let us consider that there is only one value of acceleration or braking on one infrastructure block section during train movement. Train speed on an infrastructure block section will be in the form (2).

\[ v^i(t) = v_0^i + a^i t, \]   (2)
where \(v_0^i\) — train speed at the entering of block section number \(i\), \(a^i\) — train acceleration or braking at the infrastructure block section number \(i\). If \(a^i < 0\) we consider that train brakes, else train accelerate.

Consider that coordinate of the train “head” marks his placement at infrastructure. Then dependence of the train placement at the infrastructure block section from the time variable will be as follows (3).

\[
x^i(t) = x_0^i + v_0^i t + \frac{a^i t^2}{2},
\]

where \(x_0^i\) — the start of infrastructure block section \(i\), it must be equal to the end of previous block section. Then \(t^i = \frac{v^i - v_0^i}{a^i}\) — movement time by the infrastructure block section \(i\). If acceleration at infrastructure block section \(a^i = 0\) (uniform motion case), then \(t^i = \frac{x^i - x_0^i}{v_0^i}\).

Using these equations, modeling subsystem uniquely determines train placement at the infrastructure while evaluating the effect of different options and decisions in the scheduling scene.

### 3 SOFTWARE IMPLEMENTATION

#### 3.1 Realization features

The following time indicators for performance analysis of the multi-agent system for adaptive real-time train route management were used: infrastructure load time, train load time in an operating domain, rescheduling time for a new arrival, re-scheduling time depending on duration of maintenance window, re-scheduling time for added maintenance windows, re-scheduling time depending on the number of tracks with maintenance windows, re-scheduling time depending on speed limits on a block section, re-scheduling time depending on the number of limitations, speed on a block section, re-scheduling time depending on the number of tracks occupied due to a speed limit at a station.

A train in an operating domain has around 45 operations (operations of passing the operating domain, stop at a station or passing a station), every train operation has its own agent, and there are around 800 trains in total. Overall are around 36000 agents. Apart from that, there are around 800 train agents, 49 station agents, 500 station route agents, 3700 block section agents and around 100-200 maintenance request and availability agents. Dividing this many agents into levels and grouping them into isolated swarms of agents, which are active at certain points of time, allows to increase the system performance [12].

In order to get implementation features for each index, an average value was defined, based on the results of 10 experiments for two operating domains: Saint-Petersburg – Buslovskaya and Saint-Petersburg – Moscow. Cumulative decisions instead of ones made by separate schedulers have been taken into consideration.

Figures relevant for planning the features of the operating domains are represented in Table 1. The Moscow – Saint-Petersburg operating domain has 2.8 times more infrastructure objects than the Saint-Petersburg – Buslovskaya operating domain.

<table>
<thead>
<tr>
<th>Operating domain</th>
<th>Number of stations</th>
<th>Number of infrastructure objects</th>
<th>Number of turnouts</th>
<th>Average preparation time of infrastructure (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Saint-Petersburg – Buslovskaya</td>
<td>17</td>
<td>1293</td>
<td>133</td>
<td>1452</td>
</tr>
</tbody>
</table>
It should be noted that the load time of the Moscow – Saint-Petersburg operating domain infrastructure is about 4.9 times higher than on the Saint-Petersburg – Buslovskaya operating domain. It is caused by building of the infrastructure model within the system and special infrastructure station agents, station limits, block sections.

Table 2 represents the change of scheduling characteristics depending on the number of scheduled trains in the operating domains. According to the Table, the difference between the scheduling times on the operating domains with around the same number of trains is 2.6 times, which is close to the difference between the amounts of infrastructure elements. Thus, dependency can be observed, which is close to linear, between the number of infrastructure elements and the scheduling time of trains on an operating domain. Rescheduling time is 15-40% less in comparison with initial scheduling time. It is caused by adaptive rescheduling based on incoming events instead of rescheduling everything from scratch. Increasing the number of trains increases the train scheduling time in direct proportion.

Table 2. Planning scheduling characteristics depending on the number of tasks.

<table>
<thead>
<tr>
<th>Operating domain</th>
<th>Number of trains</th>
<th>Average scheduling time (ms)</th>
<th>Average re-scheduling time (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Saint-Petersburg – Buslovskaya</td>
<td>12</td>
<td>1511</td>
<td>1185</td>
</tr>
<tr>
<td></td>
<td>17</td>
<td>2180</td>
<td>1429</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>2257</td>
<td>1799</td>
</tr>
<tr>
<td></td>
<td>49</td>
<td>2300</td>
<td>2092</td>
</tr>
<tr>
<td></td>
<td>67</td>
<td>2746</td>
<td>2356</td>
</tr>
<tr>
<td>Saint-Petersburg – Moscow</td>
<td>71</td>
<td>7202</td>
<td>5063</td>
</tr>
<tr>
<td></td>
<td>86</td>
<td>8786</td>
<td>6164</td>
</tr>
<tr>
<td></td>
<td>152</td>
<td>12691</td>
<td>11420</td>
</tr>
<tr>
<td></td>
<td>241</td>
<td>22453</td>
<td>22899</td>
</tr>
<tr>
<td></td>
<td>311</td>
<td>36143</td>
<td>37351</td>
</tr>
</tbody>
</table>

Table 3 represents time scheduling characteristics depending on the density of disruptions. Increasing density increases the scheduling time.

Table 3. Time scheduling characteristics depending on the density of disruptions.

<table>
<thead>
<tr>
<th>Operating domain</th>
<th>Density of disruptions</th>
<th>Average scheduling time (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Saint-Petersburg – Buslovskaya</td>
<td>No disruptions</td>
<td>2356</td>
</tr>
<tr>
<td></td>
<td>Very few disruptions</td>
<td>2476</td>
</tr>
<tr>
<td></td>
<td>Few disruptions of the same type</td>
<td>2505</td>
</tr>
<tr>
<td></td>
<td>Many disruptions of different types</td>
<td>2785</td>
</tr>
<tr>
<td>Saint-Petersburg – Moscow</td>
<td>No disruptions</td>
<td>37351</td>
</tr>
<tr>
<td></td>
<td>Very few disruptions</td>
<td>39631</td>
</tr>
</tbody>
</table>
According to the table, the scheduling time mostly depends on the number of infrastructure elements and the number of scheduled trains. Other factors such as disruptions, their number, duration and density have less influence on the total scheduling time. This can be explained by partial (adaptive) scheduling according to incoming disruptive events instead of complete rescheduling. Thus, one can speak about guaranteed time for decision-making in given infrastructure and given number of trains.

The following qualitative characteristics can be noted: no thickening of lines on graphics of train routes, no unjustified changing of tracks, no traffic jams between stations, keeping security intervals, almost no delays among intercity and high-speed trains in conflict situations, average train delays less than 9% (20 trains engaged in one conflict).

This outcome has been achieved on such large-scale planning tasks for the first time.

3.2 Examples of resolved conflict situations

Let us consider a situation with a high number of disruptions as shown in Figure 2, with 6 maintenance windows, two of which completely block the traffic between Roshino and Zelenogorsk for an hour.

![Figure 2](image-url)

**Figure 2.** Bypassing 6 maintenance windows, resuming after the disruptions have been eliminated.

Resolving this situation required involving back links between different planning levels. Due to a high number of thickened graphic lines after the window the trajectory subsystem was unable to create the final schedule, since the traffic security requirements didn’t allow the trains to stop and switch on to the alternative route. The path scheduler registers this mismatch as a conflict and sends a message to the time scheduler. In order to resolve the situation, the time subsystem must delay a few trains from previous stations (for example, 2048, 6155, 6163), taking the overload of the station limits into account. It sends the newly made decision to the path scheduler. It builds the route and checks the decision to satisfy traffic security requirements. As a result, the schedule has turned out to be more balanced and stable to possible further disruptions, the effect of the maintenance window has been localized, after which the schedules tend to be exemplary again.
An overload between two stations cannot be resolved without the time scheduler, because changing the route with switching on to the opposite tracks will cost much more than changing the train schedule.

4 CONCLUSIONS

The suggested network-centric system of adaptive train scheduling based on multi-agent technologies have been developed within the project of the unified intelligent train scheduling system for the Russian railways and now is in production usage [16, 17]. The expected results of the developed adaptive train scheduling system include: reduced reaction time, increased flexibility and quickness of decision-making in response to disruptive events, increased effectivity of railway resource management in real time and securing on-time performance of trains, reduced man-hours for rescheduling trains, a completely new intelligent software system for traffic management in real time. The mentioned solutions will help increase the quality of decision-making and performance level of end-users.

The project has been funded by the Russian Foundation for Basic Research, the Ministry of Education and Science of Russia under the SSAU “Increase of competitiveness among the worldwide leading research and education centers” program for 2013-2020.

REFERENCES


Numerical and Experimental Study on Contact Force Fluctuation between Wheel and Rail Considering Rail Flexibility and Track Condition

Saki IENAGA*, Yoshiaki TERUMICHI#, Kazuhiko NISHIMURA†, Minoru NISHINA‡

* Department of Science and Technology, Graduate School of Sophia University 7-1 Kioi-cho, Chiyoda-ku, Tokyo 102-8554, Japan
saki.ienaga@sophia.ac.jp

# Department of Science and Applied Science, Sophia University 7-1 Kioi-cho, Chiyoda-ku, Tokyo 102-8554, Japan
y-terumi@sophia.ac.jp

† Technology Research and Development Department (KOMAKI), Central Japan Railway Company 1545-33 Ohyama, Komaki-shi, Aichi 485-0801, Japan [kazuhiko.nishimura, minoru.nishina]@jr-central.co.jp

ABSTRACT

This study proposes an analytical model with a multibody system considering three-dimensional wheel/rail contact geometry and ballasted track characteristics. The suppression of the contact force fluctuation between wheel and rail is desirable from the viewpoint of ensuring running safety, track maintenance, riding comfort, and minimizing impacts such as noise on surrounding environments. In this paper, we investigate the effects of the support characteristics of ballasted track on the interaction between vehicles and tracks. Numerical simulations and experiments were carried out for railway vehicle motion under a wide range of rigidities of ballasted track. Using the proposed numerical model, we obtained analysis results that are consistent with experimental results under two track conditions, one that simulates the regular ballasted track characteristics and one that has sufficient space provided between sleepers and ballasts. The proposed numerical simulation accurately analyzed vehicle motion running over ballasted track by considering the interaction between the vehicle and the track.

1 INTRODUCTION

Railways have spread throughout the world as a transportation system that has little impact on the environment. Numerous studies have researched the running safety and ride comfort of railways. These studies have reported on the issues of coupled vibration of the vehicle/track motion from the contact between wheels and rails [1].

In particular, suppressing fluctuations in the contact force between wheels and rails leads to reduced oscillation of vehicles and reduced dynamic loads of the tracks. However, it requires a great deal of effort to continuously observe the contact force fluctuations generated during the running of railway vehicles and to derive the vibration characteristics related to the vehicle and track state. Interpreting the measurement results on the motion of vehicles is not easy because there are many types of fluctuation factors, the vehicle running conditions are not always stable, and limitations exist in performing frequency analyses of the transient response. In contrast, numerical simulations enable us to freely set the conditions of vehicles and tracks and to understand their behavior continuosly and simultaneously [2][3]. Therefore, numerical simulations are an effective method of clarifying the mechanism that gives rise to fluctuations in the wheel/rail contact force. To analyze the complex motion in the vehicle/track system numerically, modeling and formulation with high accuracy are important.

In this study, we investigate the mechanism of contact force fluctuations in both numerical and experimental approaches. The modeling and formulation of a vehicle and track system are performed with consideration of the three-dimensional wheel/rail contact geometry and the...
flexibility of the track from the viewpoint of multibody dynamics. Numerical simulations using the proposed model are carried out for vehicle motion under various track-support conditions. Experiments are performed under the same condition and the results are used to improve the accuracy of the numerical simulation results.

2 MODELING AND FORMULATION

The modeling and formulation for a railway vehicle and track are discussed in this section. To construct a multibody system considering three-dimensional motion, the equations of motion are described without simplifications, such as a lack of in-plane motion calculations or the linearization of angular displacements. We propose a vehicle/track model that includes the wheel/rail contact model shown in Figure 1.

2.1 Vehicle Model

The vehicle model is shown in Figure 2. The vehicle model consists of one car body, two bogie frames, and four wheelsets with two wheels in each wheelset. These are regarded as rigid bodies connected by springs and dampers. Thus, the vehicle model is a coupled vibration system of seven rigid bodies. Each body has six degrees of freedom (DOFs): longitudinal displacement $X'$, lateral displacement $Y'$, vertical displacement $Z'$, yaw angle $\psi'$, roll angle $\phi'$, and pitch angle $\theta'$. Angles $\phi'$, $\theta'$, and $\psi'$ are rotational angles about the body coordinate axes $x'$, $y'$, and $z'$, which originate at the center of gravity of body $i$. The superscript indicates each body: $i = C, B1, B2, W1, W2, W3, W4$, where C, B, and W refer to the car body, the bogie frame, and the wheelset, respectively. The connecting elements between the bogie frame and the wheelset are eight primary suspensions, each consisting of three springs ($k_{1X}$, $k_{1Y}$, $k_{1Z}$) and three dampers ($c_{1X}$, $c_{1Y}$, $c_{1Z}$). The connecting elements between the car body and the bogie frame are four air suspensions, each consisting of three springs ($k_{2X}$, $k_{2Y}$, $k_{2Z}$) and a damper ($c_{2}$); two lateral stoppers and dampers, each consisting of a spring ($k_{3X}$) and a damper ($c_{3X}$); four yaw dampers ($c_{4}$); and two traction devices, each consisting of two springs ($k_{5X}$, $k_{5Z}$). The subscripts $X$, $Y$, and $Z$ indicate the longitudinal, lateral, and vertical directions, respectively. The forces of the lateral stopper ($k_{3X}$) and the vertical spring of the traction device ($k_{5Z}$) act as extra forces only if the displacement of the car body relative to the bogie frame exceeds the clearances. To account for the strong nonlinearities of the springs caused by large vehicle motions, modeling for the nonlinearities is introduced as follows:

(a) with respect to the relative motion of the bogie frame and the wheelset, the longitudinal and lateral stiffness of the primary suspension increases ($k_{1X}$, $k_{1Y}$) if the relative displacement becomes large due to the nonlinearity in the laminated rubber in the suspension, and the vertical
stiffness of the primary suspension increases \((k_{1Z})\) if the vertical relative displacement is larger than the clearance due to the stopper between the bogie frame and the wheelset;

(b) with respect to the relative motion of the car body and the bogie frame, the vertical stiffness of the air suspension increases \((k_{2Zs})\) if the downward relative displacement is larger than the clearance inside the air suspension.

Figure 2. Vehicle model

2.2 Track Model

The track model is shown in Figure 3. The track model consists of rails, sleepers, and ballasts connected to each other by springs and dampers. In the track model, the rails are treated as flexible beams by applying an absolute nodal coordinate formulation (ANCF) \([5]\). In the ANCF, the global position vector of an arbitrary point on the element can be described by using the global shape function and the nodal coordinates as follows:

\[
\mathbf{r} = \mathbf{S}(x, y, z) \mathbf{e}
\]  

where \(\mathbf{S}\) is the global shape function and \(\mathbf{e}\) is the vector of the element nodal coordinates. The global shape function \(\mathbf{S}\) is given by

\[
\mathbf{S} = \begin{bmatrix} S_1 I & S_2 I & S_3 I & S_4 I & S_5 I & S_6 I & S_7 I & S_8 I \end{bmatrix}
\]

\[
S_1 = 1 - 3\xi^2 + 2\xi^3, \quad S_2 = l_e (\xi - 2\xi^2 + \xi^3), \quad S_3 = l_e (\eta - 3\xi^2), \quad S_4 = l_e (\xi - 3\xi^2),
\]

\[
S_5 = 3\xi^2 - 2\xi^3, \quad S_6 = l_e (-\xi^2 + 2\xi^3), \quad S_7 = l_e \xi, \quad S_8 = l_e \xi^2
\]

where \(\xi = x/l_e, \eta = y/l_e, \gamma = z/l_e\), \(x, y, z\) are the displacements of an arbitrary point from the origin in the undeformed reference configuration, and \(l_e\) is the length of the element. The nodal coordinates are defined as nodal displacements and slopes that are determined in the undeformed reference configuration as follows:
\[ e = \begin{bmatrix} \mathbf{r}^T \\ (\partial \mathbf{r}/\partial x)^T \\ (\partial \mathbf{r}/\partial y)^T \\ (\partial \mathbf{r}/\partial z)^T \end{bmatrix} \]  

(11)

This method leads to a constant mass matrix derived from the kinetic energy of the element and a nonlinear stiffness matrix derived from the strain energy.

The rails are supported by concrete sleepers placed at specific intervals. The sleepers are regarded as rigid bodies and each has three DOFs: lateral displacement \( Y_{sl} \), vertical displacement \( Z_{sl} \), and roll angle \( \phi_{sl} \). They are connected to the rails and ballasts elastically. The ballasts are regarded as lumped masses with vertical displacement \( Z_{ba} \) and they form the three layers connected to the sleepers and the subgrade elastically.

![Figure 3. Track model](image)

**2.3 Track Model under a Wide Range of Support Rigidities**

To investigate the effect of support rigidity of the ballasted track, a numerical analysis is performed for the vehicle running over the track with extreme changes of the support stiffness or damping. As shown in Figure 4, we simulate the following three states of ballasted track support.

State A: The track has regular support rigidity.

State B: The boundary area between A and C has lower support stiffness and damping than that in A.

State C: The support stiffness and damping are removed in a certain range.

![Figure 4. Test track model with unsupported sleepers](image)

At the C section, where the sleepers are completely unsupported by ballasts, track rigidity is decreased. This leads to track irregularity because rail deformation increases due to the cyclic loadings of the running vehicles. At the B section, before and after the C section, rails are deformed in the upper direction and a gap is generated between sleepers and ballasts, as shown in Figure 5. In this model, the B section is the length equivalent to three sleepers and the support force at the section simulates the gap due to the track irregularity and the decreased track rigidity.
rigidity at the boundary. Therefore, the support force between sleepers and ballasts \( F_{sz} \) and the support force among ballasts \( F_{bz} \) are formalized as follows:

(a) The support force does not act unless the rail displacement reaches gap size \( z_d \).

(b) The support stiffness and damping coefficients are set to values smaller than those in the A section.

\[
F_{sz} = \begin{cases} 
0 & \text{if } \left| z_s - z_{ba} \right| < z_d \\
C_B \cdot k_{sz} \left( (z_s - z_{ba}) - z_d \right) + C_B \cdot c_{sz} \left( \dot{z}_s - \dot{z}_{ba} \right) & \text{if } z_s - z_{ba} \geq z_d \\
C_B \cdot k_{sz} \left( (z_s - z_{ba}) + z_d \right) + C_B \cdot c_{sz} \left( \dot{z}_s - \dot{z}_{ba} \right) & \text{if } z_s - z_{ba} \leq -z_d 
\end{cases}
\]

\[
F_{bz} = \begin{cases} 
0 & \text{if } \left| z_{ba} \right| < z_d \\
C_B \cdot k_{bz} \left( z_{ba} - z_d \right) + C_B \cdot c_{bz} \dot{z}_{ba} & \text{if } z_{ba} \geq z_d \\
C_B \cdot k_{bz} \left( z_{ba} + z_d \right) + C_B \cdot c_{bz} \dot{z}_{ba} & \text{if } z_{ba} \leq -z_d 
\end{cases}
\]

where \( k_{sz}, c_{sz} \) are the stiffness and damping coefficients between the sleeper and the ballast, respectively; \( k_{bz}, c_{bz} \) are the stiffness and damping coefficients among the ballasts and \( z_s \), respectively; and \( z_{ba} \) is the vertical displacement of the sleeper and the ballast. \( C_B \) is the coefficient of the decrease of the support stiffness and damping and is set to 0.5, and \( z_d \) is the length of the gap between sleepers and ballasts at the B section and is set to 1.0 mm.

![Figure 5. Track irregularity around the C section](image)

**2.4 Wheel/Rail Contact Model** [6][7][8]

The normal contact force between the wheel and the rail is defined by using Hertz’s contact theory as the elastic contact model. The normal contact force is defined by the amount of elastic deformation of wheel and rail, \( \delta_n \). The normal contact force is given as follows:

\[
N = -K_c \delta_n^{3/2} - C_c \delta_n \left| \dot{\delta}_n \right|
\]

where \( K_c \) is the Hertzian contact stiffness and \( C_c \) is the damping coefficient. The damping force is modified by including the factor \( \left| \dot{\delta}_n \right| \) so that the contact force is zero when the indentation is zero. The amount of elastic deformation between wheel and rail \( \delta_n \) is given as

\[
\delta_n = \vec{n}^R \cdot \left( \vec{r}_P^W - \vec{r}_P^R \right)
\]

where \( \vec{r}_P^W \) is the global position vector of the contact point on the wheel, \( \vec{r}_P^R \) is the global position vector of the contact point on the rail, and \( \vec{n}^R \) is the normal unit vector at the contact point on the rail. Tangential forces are defined as creep forces based on a FASTSIM algorithm known as Kalker’s nonlinear theory while the vehicle is in motion.

To calculate the contact force precisely, the three-dimensional contact geometry including the wheel and rail profiles should be considered. The contact geometry is modeled by surface parameters that describe the location of the contact point and define the wheel and the rail profiles. The four parameters for one contact point are as follows: \( s_1^W \) is the lateral parameter on the wheel cross-section, \( s_2^W \) is the circumferential parameter of the wheel, \( s_1^R \) is the longitudinal
parameter of the rail, and $s_2^R$ is the lateral parameter on the rail cross-section. These parameters are shown in Figure 6. By using these parameters, the rolling radius (profile of the wheel cross-section) can be written as a function of $s_1^W$, $f^W(s_1^W)$, and the profile of the rail cross-section can be written as a function of $s_2^R$, $f^R(s_2^R)$.

$$s = \begin{bmatrix} s_1^W & s_1^W & s_2^R & s_2^R \end{bmatrix}^T$$

Figure 6. Surface parameters

We adopt a contact algorithm that calculates contact points between wheels and rails for every time step in the kinematic analysis. In the contact position analysis, when two rigid bodies come into contact, two types of nonconformal kinematic contact conditions need to be satisfied. First, the vector between two contact points on the two surfaces intersects the contact surface at right angles. Second, the two surfaces must have the same tangent planes at the contact point. These two conditions define the following four constraint equations required to describe the nonconformal contact between the wheel and rail:

$$E(s^W,s^R) = \begin{bmatrix} t_1^R \cdot (r^W_1 - r^R_1) \\ t_2^R \cdot (r^W_2 - r^R_2) \\ t_1^W \cdot n^R \\ t_2^W \cdot n^R \end{bmatrix} = 0$$

where $t_i^W$ and $t_i^R$ are the two tangents of the wheel, respectively, and $n^R$ is the normal to the surface of the rail at the contact point. For a given set of wheel/rail generalized coordinates, Eq. (17) is solved iteratively by the Newton-Raphson algorithm to determine the surface parameters. This iterative process continues until convergence is achieved.

### 2.5 Formulation of Motion Equations

The equations of motion for the vehicle-track system use the generalized Newton-Euler equations, which are written in matrix form as follows:

$$M^i q^C = Q^C_i + Q^C_v$$

$$M^{RB} q^{Bj} = Q^{Bj}_f + Q^{Bj}$$

$$M^{Wj} q^W = Q^{Wj}_f + Q^{Wj} + Q^{Wj}_v$$

$$M^{Bi} q^{Bi} = Q^{Bi}_f + Q^{Bi} + Q^{Wj}_v$$

$$\begin{bmatrix} C^{Bj} & e^{Bj}^T \\ e^{Bj} & \lambda^{Bj} \end{bmatrix} = \begin{bmatrix} -Q^K_i - Q^{Bi}_j - Q^{Bi}_f + Q^{Bi}_j \end{bmatrix}$$

$$M^{Si} q^{Si} = Q^{Si}_i + Q^{Si}$$

$$M^{Ba} q^{Ba} = Q^{Ba}_i - Q^{Si}_j$$

where superscript $i$ denotes the body number, $M^i$ is the generalized mass matrix, $q^i$ is the generalized coordinates, and $Q^i$ is the generalized force vector including inertia forces and
gravitational force. The generalized external force vectors are as follows: $Q_f^i$ is the spring and damper force, $Q_c^i$ is the wheel/rail contact force, $Q_k^i$ is the rail elastic force, $Q_s^i$ is the track support force, and $Q_g^i$ is the gravitational force. $C_{e^{ij}}$ is the Jacobian matrix for the constraint of the track support and $\lambda^{ij}$ is the Lagrange multiplier.

3 EXPERIMENT

In this study, railway vehicle running experiments are performed under a local change of track rigidity. The wheel/rail contact forces and rail displacements are measured when the vehicle runs over the track.

3.1 Experimental method

Figure 7 shows the setup of the experiment. The test line consists of an acceleration section, a test section, and a deceleration section to reproduce a real main line. In the test section, sufficient space is provided between the sleepers and ballast supporting rails and the total length of the space is variable. During the experiment, the test vehicle starts and accelerates by an accelerating truck and is released by decelerating the truck before the test section.

The test vehicle uses two real bogies and a car body, on which are mounted weights for adjusting the wheel load and an air tank for operating the air springs in order to reproduce real vehicle motion. However, the distance between the centers of the bogies is shorter than the actual distance in consideration of the vehicle weight and the truck acceleration performance.

In the experiments, wheel loads, lateral pressures, and vertical displacements of the rails are measured. Wheel loads and lateral pressures are obtained by converting the strains generated in each wheel. The relation between strain and load is known in advance. The vertical displacements of the rails are measured with laser displacement sensors. The sensors are installed in the foundation in such a way as to prevent them from moving with the rails; they measure the amount of displacement by laser irradiation of reflecting plates installed on the rails.

![Figure 7. Test track and test vehicle](image-url)
4 NUMERICAL SIMULATION

To investigate the effects of the support characteristics of the ballasted track, numerical simulations are carried out by using the proposed analytical model. For the purpose of validation, the numerical results are compared with those of the vehicle running experiments.

4.1 Numerical Parameters

Table 1 lists the specification parameters for the numerical simulations. The parameters are based on typical parameters for a Japanese high-speed train. We adopted the arc wheel profile of a Shinkansen and a 60 kg rail profile, both of which are used in the profile functions.

<table>
<thead>
<tr>
<th>Table 1. Numerical parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter</td>
</tr>
<tr>
<td>Car body mass $m_C$</td>
</tr>
<tr>
<td>Bogie frame mass $m_B$</td>
</tr>
<tr>
<td>Wheelset mass $m_W$</td>
</tr>
<tr>
<td>Longitudinal direction of axle box suspension on bogie frame $a_{1b}$</td>
</tr>
<tr>
<td>Longitudinal direction of air spring on car body $a_{2c}$</td>
</tr>
<tr>
<td>Rail mass per unit length $m_r$</td>
</tr>
<tr>
<td>Sleeper mass $m_S$</td>
</tr>
<tr>
<td>Inertia moment of sleeper $I_S$</td>
</tr>
<tr>
<td>Upper ballast $m_{Ba1}$</td>
</tr>
<tr>
<td>Middle ballast $m_{Ba2}$</td>
</tr>
<tr>
<td>Lower ballast $m_{Ba3}$</td>
</tr>
<tr>
<td>Support stiffness of rail $(k_{rY}, k_{rZ})$</td>
</tr>
<tr>
<td>Support damping coefficient of rail $(c_{rY}, c_{rZ})$</td>
</tr>
<tr>
<td>Support stiffness of rail $k_{sY}$</td>
</tr>
<tr>
<td>Support stiffness of rail $k_{sZ}$</td>
</tr>
<tr>
<td>Support damping coefficient of rail $c_{sY}$</td>
</tr>
<tr>
<td>Support damping coefficient of rail $c_{sZ}$</td>
</tr>
<tr>
<td>Support stiffness of rail $(k_{b1Z}, k_{b2Z})$</td>
</tr>
<tr>
<td>Support damping coefficient of rail $(c_{b1Z}, c_{b2Z})$</td>
</tr>
<tr>
<td>Support stiffness of rail $k_{b3Z}$</td>
</tr>
<tr>
<td>Support damping coefficient of rail $c_{b3Z}$</td>
</tr>
<tr>
<td>Interval of sleeper $l_s$</td>
</tr>
<tr>
<td>Length of track $l_T$</td>
</tr>
<tr>
<td>Number of elements of a rail $n_e$</td>
</tr>
<tr>
<td>Track gauge $2L_R$</td>
</tr>
</tbody>
</table>

4.2 Effect of Track Rigidity on Rail Displacement

The effect of the support characteristics of ballasted track on the rail displacement is discussed in this section.

Figure 8 shows numerical and experimental results of the vertical displacements of the rail under two support conditions of the track: (a) when the vehicle is passing over the A section at 40 km/h, (b) when the vehicle is passing at 45 km/h over the C section where five sleepers are unsupported by ballasts. Under both conditions, we obtained numerical results that are consistent with the experimental results shown in Figure 8. Since the track rigidity at the C section is lower than that at the A section, the following analysis results are obtained: the rail is displaced by about 0.7 mm when the wheelsets pass over the A section (Figure 8(a)). On the other hand, the rail is displaced by about 6 mm when the front and rear bogies pass over the C section (Figure 8(b)). Furthermore, during the front bogie and the rear bogie passes, the rail displacement is restored to the original position at the A section (Figure 8(a)). On the other hand, the rail displacement produced by the front bogie passing is not completely restored at the C section (Figure 8(b)).
Figure 8. Vertical displacements of the rail

(a) Passing over the A section at 40 km/h  
(b) Passing over the C section at 45 km/h

Figure 9 shows three positions of the 1st wheelset on the track; Positions I, II, and III indicate when the 1st wheelset is located on the No. 1, No. 3, and No. 5 sleepers, respectively. Figure 10 shows the rail displacement on each sleeper in positions I, II, and III.

We can see the deformation process of the rail as the vehicle travels in Figure 10. At the C section, the rail displacements are about 10 times larger than those at the A section, and the deformation shapes of the rail are similar for each position because of the decreased track rigidity.

Although the displacements differ between the numerical and experimental results, due to track irregularities, they are in good agreement qualitatively.

Figure 9. Three positions of 1st wheelset on track

(a) Passing over the A section at 40 km/h  
(b) Passing over the C section at 45 km/h

Figure 10. Vertical displacements of rail arrangements I, II, and III
4.3 Effect of Track Rigidity on Contact Force

The effect of the support characteristics of ballasted track on the vertical contact force, namely wheel load, is discussed in this section.

Figure 11 shows the numerical and experimental results of the wheel load when the vehicle is passing over the A section with regular support rigidity and the C section where three or five sleepers are unsupported by the ballasts. In the numerical simulations, the vehicle speed is set to be the same as in the experiments: 20 km/h, 25 km/h, 40 km/h, and 45 km/h. Though high frequencies associated with measurement noises and track irregularities are seen in the experimental results, the wheel load fluctuation caused by the change of track rigidity at the C section is observed in the numerical results, as well as in the experimental results. The details of each analytical condition are described as follows.

In the experimental results shown in Figure 11(a), not much change is seen in the wheel load caused by the decreased track rigidity at the C section. This is because the wheel load fluctuates dominantly depending on the track irregularities at low speeds such as 20 km/h and 25 km/h. In the numerical results shown in Figure 11(b), the wheel load fluctuates immediately after passing at the C section. However, the amounts of fluctuation in the numerical results are smaller than those in the experimental results because of track irregularities.

In both experimental and numerical results shown in Figures 11(c) and (d), we can observe that wheel load fluctuations are caused by the decreased track rigidity at the C section. The reason is that the vehicle performs a pitching motion when it passes the C section. The amount of wheel load fluctuation after the C section increases. The phase is delayed as the number of consecutive unsupported sleepers increases from three to five. This result indicates that the pitch angle fluctuation produced by passing the C section increases with the increase of the number of unsupported sleepers.

![Figure 11. Wheel loads](image)
Figures 12(a)–(d) show the vehicle motion around the C section. When the 1st wheelset reaches the C section, the vehicle starts pitching with downward motion due to the difference of the track rigidity at the wheel positions. As the pitch angle increases, the wheel load on the 1st wheelset increases, as shown in Figure 11. After the 1st wheelset passes the B section after the C section, the pitch angle of the vehicle becomes the maximum and the wheel load becomes the maximum (Figure 12(c)).

(a) The 1st wheelset has reached the C section
(b) The 1st wheelset is in the middle of the C section
(c) The 1st wheelset has passed the C section, but the 2nd wheelset has not
(d) Immediately after the front bogie has passes the C section

Figure 12. Vehicle motion around the C section

5 CONCLUSIONS

This study presented a vehicle/track model that takes into account the three-dimensional wheel/rail contact geometry and the ballasted track characteristics. The purpose is to elucidate the effects of the support characteristics of ballasted track on the interaction between the vehicle and the track. Numerical simulations were conducted for a railway vehicle running under the conditions that include a change of track support rigidity and experiments were conducted in the low-speed range under the same conditions. The following conclusions were obtained for the speed region analyzed in this study.

(1) Numerical simulations of the interaction between vehicle and track when a vehicle passes can be performed with high accuracy. Good agreement is obtained for the proposed model simulations and the experiments under both track conditions, one that simulates the regular ballasted track characteristics and one that has sufficient space provided between sleepers and ballasts.

(2) An increase occurs in the amount of rail displacements when the vehicle passes over the section with decreased track support rigidity. The displacements are not restored completely to their former state between front and rear bogie passes.

(3) Although a significant fluctuation in wheel load does not occur when the vehicle passes over the section with the decreased track support rigidity, a rapid wheel load increase is caused by the pitching of the vehicle after passing the section. The amount of wheel load fluctuation increases as the length of the section increases because the pitch angle is larger.

It is demonstrated that the vehicle motion when running over ballasted track that has a change in support rigidity is analyzed with high accuracy by using the numerical simulation model proposed in this study.
REFERENCES


Partially-linearized multibody equations of railroad vehicles on arbitrary tracks for on-board applications

Javier F. Aceituno, José L. Escalona, Daniel García-Vallejo

Department of Mechanical and Manufacturing Engineering
University of Seville
Camino de los Descubrimientos s/n, 41092, Sevilla, Spain
[aceituno, escalona, dgvallejo]@us.es

ABSTRACT

This paper shows a partial linearization of multibody equations of motion applied to multibody models of railroad vehicles for on-board simulations on arbitrary tracks. The equations of motion are first symbolically obtained as a transformation of the Newton-Euler equations of the vehicle bodies that are referred to a track frame that accompanies the vehicle to then, numerically be linearized respect to equilibrium positions obtained at different stretches, such as tangent and constant radius curve ones. Subsequently, if a vehicle body runs in a transition stretch during the dynamic simulations, precalculated mass, damping and stiffness matrices are interpolated. In addition, wheel-rail contact constraints are treated with precalculated lookup tables which can take into account the track irregularities. A wheelset case study is presented showing computational efficiency and accuracy in the results.

Keywords: Multibody systems, railroad vehicles, linearization, contact look-up tables.

1 INTRODUCTION

In railroad dynamics, an accurate prediction of the wheel-rail contact geometry is one of the most essential issues for evaluating vehicle stability, curve negotiations or ride comfort [2]. The solution of wheel-rail contact problem is required for an accurate prediction of the position of the contact points and contact forces between wheel and rail [8]. In addition, if this prediction were an on-board one, a great reduction of railroad maintenance costs could be achieved allowing every vehicle body the possibility to check online track irregularities.

Multibody equations of motion of mechanical systems are highly nonlinear and require an important computational effort to be solved in time. Many scientific research has been done on linearization of these equations of motion in order to achieve computationally efficient simulations. In [9] an approach to linearize nonlinear Differential Algebraic Equations of index 3 is proposed by augmenting the governing equations of the system to then, compute a set of quantities of interest that provide the linearization. In [11], linearized equations of motion of a Whipple bicycle model are presented where for benchmarks purposes, accurate eigenvalues and the speeds at which the model is self-stable are calculated. Also, in [12], a numeric technique for generating equations of motion using Lagrange’s method is developed for vehicle stability analysis.

Together with the linearization of the multibody equations of motion, the symbolic computation is another source of research in order to achieve efficient multibody dynamic simulations. This is, to obtain explicit expressions of the equations of motion. In [13], the symbolic generation of multibody systems applied to the field of identification, modeling and analysis that can include constraints resolution and system reduction is presented. Fisette and Samin also gathered the state of the art of symbolic modeling of multibody system in [14]. In addition, applications where the symbolic calculation is applied to linearization procedures can be seen in [10] and in [3]. In [10], the equations of motions of constrained multibody systems obtained by Kane’s method are symbolically linearized after a previous organization. This method is applied to a Whipple bicycle model presented in [11] obtaining accurate results. In [3], the equations of motion applied...
to railroad vehicles are symbolically obtained, and their resolution is speed up with the use of precalculated contact look-up tables.

In this work, a symbolic generation of the Newton-Euler equations of motion applied to railroad vehicles is first developed before the partial linearization. To this end, the paper is divided as follows: in Section 2 the kinematic description of railroad vehicles is presented where the different reference frames and terminology used are described. Section 3 presents the multibody dynamic equations of motion of railroad vehicles together with a brief description of the wheel-rail contact look-up tables. Then, in Section 4 the equations of motion are expressed in terms of the independent coordinates to be numerically linearized as Taylor series. Also the procedure followed to interpolate between the different linearized matrices is described. Afterwards, the wheelset case study is presented in Section 5 and its numerical results and discussion are shown in Section 6. Finally, Section 7 gives a summary and conclusions.

2 KINEMATIC MODELLING OF RAILROAD VEHICLES

In this work, the equations of motion of the vehicle bodies are referred to a track frame that accompanies the vehicle [1]. Equation (1) states that the global position vector \( R_{ip} \) of an arbitrary point \( P \) of body \( i \) can be calculated as

\[
R_{ip} = R' + A' \bar{r}_p
\]

where \( R' \) is the global position vector of the track frame, \( A' \) is the rotation matrix and \( \bar{r}_p \) is the local position vector of point \( P \). However, for the kinematic description of wheelset bodies, five different reference frames are needed as can be seen in Fig. 1. These frames are; the global frame (GF), track Frame (TF), wheelset-track frame (WTF), wheelset intermediate frame (WIF) and body frame (BF).

![Figure 1: Wheelset kinematic description.](image)

2.1 Reference frames

The TF \([O'(X'Y'Z')]\) is assumed to follow the track center line keeping axis \( X' \) tangent to the track centerline. The arc length along the track centerline traveled by the TF is denoted as \( s' \). The position and orientation coordinates that describe the TF respect to the GF are \( q' = [(x' y' z' \phi' \theta' \psi')] \).
where a set of three consecutive rotations applied to the GF is used starting with the yaw angle $\psi$ about the positive direction of axis $Z$, followed by a pitch rotation $\theta$ about the negative direction defined by the rotated axis $Y'$, followed by a roll rotation $\phi$ about the negative direction defined by the rotated axis $X''$. These coordinates are known functions of the track centerline geometry as $q_{t'} = q_t(s')$ using a rail preprocessor.

The wheelset frame or body frame BF $[O_w(X_wY_wZ_w)]$, whose coordinates respect to the track frame are $q_w = [(x_w y_w z_w \phi_w \theta_w \psi_w)]$ where the three consecutive rotations applied to the TF to define its orientation starts with the yaw angle $\psi_w$ about the positive direction of axis $Z'$, followed by a roll rotation $\phi_w$ about the positive direction of the rotated axis $X'$, and followed by a pitch rotation $\theta_w$ about the positive direction of the rotated axis $Y''$. In this case it is convenient to leave the pitch rotation as the last one.

The WIF $[O_{wl}(X_{wl}Y_{wl}Z_{wl})]$, whose coordinates with respect to the TF are $q_{wl} = [(x_{wl} y_{wl} z_{wl} \phi_{wl} \theta_{wl} \psi_{wl})]$. This is, the BF is the result of rotating the WIF the pitch angle $\theta_w$ about the common $Y_w$ axis.

The WTF $[O_{wt}(X_{wt}Y_{wt}Z_{wt})]$ that accompanies the wheelset along the track centerline. It is a convenient frame for the numerical treatment of the wheel-rail contact with look-up tables as it will be shown in Section 4. Its coordinates respect to the GF are known functions of the parameter $s_{wt}$ that has to be calculated solving the following non-linear algebraic equation:

$$i_{tcl}(s_{wt})^T(R_w - R_{tcl}(s_{wt})) = 0 \tag{2}$$

where $R_{tcl}(s)$ and $i_{tcl}(s)$ are functions that provide the position and tangent vector of a point in the track centerline given the arc length $s$, and $R_w$ is the global position vector of the wheelset.

Equation (2) states that the position vector of the WF with respect to the WTF is perpendicular to the tangent to the track at the body location. In other words, it states that the position vector has zero $X$ coordinate in the WTF.

### 2.2 Vector components

In what follows, the symbols that are used to express the vector components in the different frames, are identified:

- **Bold symbols without superscript**, like $\mathbf{R}$, means the 3x1 column matrix that contains the component of vector $\vec{R}$ in the GF.
- **Bold symbols with 'bar' superscript**, like $\overline{\mathbf{r}}$, means the 3x1 column matrix that contains the components of vector $\vec{r}$ in the TF.
- **Bold symbols with 'hat' superscript**, like $\mathbf{\hat{r}}$, means the 3x1 column matrix that contains the components of vector $\vec{r}$ in the BF.
- **Bold symbols with 'arc' superscript**, like $\mathbf{\overset{\ddot{}}{r}}$, means the 3x1 column matrix that contains the components of vector $\vec{r}$ in the WTF.
- **Bold symbols with 'inverted arc' superscript**, like $\mathbf{\overset{\ddot{}}{r}}$, means the 3x1 column matrix that contains the components of vector $\vec{r}$ in the WIF.

### 3 DYNAMICS MODELLING OF RAILROAD VEHICLES

In this section, the equations of motion based on the Newton-Euler equations of the rigid bodies are first obtained for unconstrained vehicles. Then the wheel-rail contact with look-up tables is briefly discussed to present the augmented equations of motion as a system of differential algebraic equations (DAE).
3.1 Newton-Euler equations for vehicle bodies

On the one hand, Newton equations for vehicle bodies are projected to the TF as:

\[ m_i \dddot{\bar{R}}_i = \bar{F}_i \quad (3) \]

where \( \dddot{\bar{R}}_i \) is the acceleration of the center of gravity of body \( i \), \( m_i \) the body mass and \( \bar{F}_i \) the sum of all forces applied to the body projected to the TF.

On the other hand, Euler equations can be projected to the BF for non-wheelset bodies (superscript \( nw \) and 'hat' symbol) or to the WIF for wheelset bodies (superscript \( w \) and 'inverted arc' symbol).

For simplicity, in the case of wheelset bodies these equations take the form:

\[ I_w \dddot{\bar{\alpha}}_w = \ddot{\bar{T}}_w - \bar{\omega}_w \times (I_w \dddot{\bar{\omega}}_w) \quad (4) \]

where \( \dddot{\bar{\alpha}}_w \) and \( \dddot{\bar{\omega}}_w \) are the angular acceleration and angular velocity respectively, \( I_w \) is the inertia tensor of the wheelset body in the WIF and \( \ddot{\bar{T}}_w \) is the vector sum of moments applied with respect to the center of gravity projecter to the WIF. Considering the railroad vehicle systems as unconstrained vehicles where the bodies interaction is due to spring-damper suspension elements, the Newton-Euler equations can be written in terms of the generalized coordinates \[2\] as:

\[ M \dddot{q} = Q_{ex} + Q_v + Q_{TF} + Q_{susp} \quad (5) \]

where \( Q_{ex}, Q_v, Q_{TF}, \) and \( Q_{susp} \) are the vector of generalized forces, the vector of quadratic velocity inertia forces, the vector of inertia forces due to the TF motion and vector of generalized suspension element forces respectively. The total mass matrix and each of the force vectors are defined as follows:

\[ M = \begin{bmatrix} M' & M'^{i+1} & \cdots & M'^{nb} \\ M^{i+1} & M^{i+1} & \cdots & M^{nb} \\ \vdots & \vdots & \ddots & \vdots \\ M^{nb} & M^{nb} & \cdots & M^{nb} \end{bmatrix}, \quad Q = \begin{bmatrix} Q' \\ Q'^{i+1} \\ \vdots \\ Q'^{nb} \\ Q'' \\ Q''^{i+1} \\ \vdots \\ Q''^{nb} \end{bmatrix} \quad (6) \]

where \( nb \) is the number of bodies of the system.

3.2 Wheel-rail contact with look-up tables

Wheel-rail contact is treated with precalculated look-up tables as presented in \[3\] that are based on the contact constraint formulation \[2\]. Therefore, in order to evaluate the wheel-rail contact, a set of contact constraints that allow four relative degrees of freedom of the wheelset with respect to the track are considered. The constraint vector \( C \) can be written as:

\[ C_j^C(q^w, s) = \begin{bmatrix} \bar{r}_C^w(q^w, s^w) - \bar{r}_C^w(s', s) \\ \bar{t}_C^{2c} - \bar{t}_C^w \\ \bar{n}_C^w \end{bmatrix} = \mathbf{0}, \quad j = L, R \quad (7) \]

being \( L \) and \( R \) the left and right wheel-rail contact, \( \bar{r}_C^w \) and \( \bar{r}_C^w \) the position vector of the contact point \( C \) in the wheel and rail respectively, \( \bar{t}_C^{2c} \) and \( \bar{t}_C^w \) the first and second tangent vectors at the contact point in the wheel, and \( \bar{n}_C^w \) the normal vector at the contact point in the rail. In Eq. (7) vector \( s \) contains the four surface parameters that define the wheel-rail geometry functions as in Fig. 2.

The Wheel-rail contact look-up table is created solving the 10 non-linear algebraic equations (five per each wheel-rail pair) of Eq. (7) for a set of values of the wheelset coordinates \( q^w \). Following the procedure presented in \[3\], the use of a constraint contact look-up table imposes two non-linear algebraic constraints relating the \( z^{w1} \) and \( \psi^{w1} \) coordinates of the wheelset respect to the
WIF, that can be written in terms of the generalized coordinates as \( C^{wr}(q^w) = 0 \), where \( C^{wr} \) is the constraint look-up table vector. Consequently, augmenting the resulting equations of Eq. (5) with the constraint equations yields:

\[
M(q^w)\ddot{q}^w + C^{wrT}\lambda^{wr} = Q_{ex} + Q_v + Q_{TF} + Q_{susp}
\]

where \( \lambda^{wr} \) refers to the vector of Lagrange multipliers due to the contact look-up table constraints.

4 PARTIAL LINEARIZATION OF EQUATIONS OF MOTION

In this section, the equations of motion are first obtained in terms of the independent coordinates to then, partially linearized respect to different equilibrium positions.

4.1 Equation of motion in terms of independent coordinates

When the equations of motion are augmented with a constraint vector as in Eq. (8), they can be easily written in terms of the independent coordinates \( q^{ind} \) if the constraint equations are not explicit function of time (scleronomic constraints).

As the look-up table constraint vector \( C^{wr} \) can be written as \( C^{wr}(q^w, \ell, \omega^w) = 0 \), the time derivative of the coordinates \( \dot{q}^w \) can be written as \( \dot{q}^w = -C_q^{-1}C_t \), where \( C_q \) and \( C_t \) are the Jacobian matrix and the partial time derivative of the constraints respectively. Identifying the independent and dependent coordinates \[4\], it yields:

\[
C_q^{ind}q^{ind} + C_q^{dep}q^{dep} = -C_t \rightarrow \dot{q}^{dep} = -C_q^{dep-1}C_q^{ind}q^{ind} + C_t
\]

where the independent and dependent wheelset coordinates are \( q^{ind} = [x^w, y^w, \theta^w, \psi^w]^T \) and \( q^{dep} = [z^w, \varphi^w]^T \) respectively. Rearranging the time derivative of the coordinates, one has the matrix \( B \) that relates \( q \) with \( q^{ind} \) as follows:

\[
\dot{q} = \begin{bmatrix} \dot{q}^{ind} \\ \dot{q}^{dep} \end{bmatrix} = \begin{bmatrix} I \\ -C_q^{dep-1}C_q^{ind} \end{bmatrix} q^{ind} = Bq^{ind}
\]

where the scleronomic constraint property has been considered.

The time derivative of Eq. (10) requires the calculation of \( B \). To this end, with the help of the time derivative of the second equation of Eq. (9) as \( \ddot{q}^{dep} = -C_q^{dep-1}C_q^{ind} \dot{q}^{ind} - C_q^{dep-1}(C_q^{ind}q^{ind} - C_q \dot{q}^{dep}) \), it can be written as:
\[
\dot{B} = \frac{\partial}{\partial t} \left[ \begin{array}{c}
 I \\
 -C^{-1}_q C_q
\end{array} \right] = \left[ \begin{array}{c}
 0 \\
 -C^{-1}_q \left( \dot{C}_q - C^{-1}_q C_q \right)
\end{array} \right]
\] (11)

Equation (11) allows to rewrite the equations of motion of Eq. (8) in terms of the independent coordinates by substituting \( \dot{q} = B\dot{q}_{\text{ind}} + \ddot{B}q_{\text{ind}} \) as follows:

\[
M \left( B\ddot{q}_{\text{ind}} + \dot{B}\dot{q}_{\text{ind}} \right) + C_{wT} \lambda_{wr} = Q
\] (12)

where for simplicity, vector \( Q \) contains all the generalized forces of the right term of Eq. (8).

Premultiplying Eq. (13) by the transpose matrix \( B^T \) and knowing that the product \( B^T C_q \) is null [4], one can write the equations of motion in terms of the independent coordinates as:

\[
B^T MB\dot{q}_{\text{ind}}^w + B^T \left( MB\ddot{q}_{\text{ind}}^w - Q \right) = 0
\] (13)

4.2 Partial linearization

The equations of motion of Eq. (13) can be written as a function vector as

\[
f(\ddot{q}_{\text{ind}}^w, \dot{q}_{\text{ind}}^w, q_{\text{ind}}^w) = 0
\] (14)

thus, neglecting the high order terms, they can numerically be linearized respect to equilibrium positions \( q_{\text{ref}} \) as:

\[
\left. \frac{\partial f}{\partial \ddot{q}_{\text{ind}}^w} \right|_{q_{\text{ref}}} (\ddot{q}_{\text{ind}}^w - \ddot{q}_{\text{ref}}) + \left. \frac{\partial f}{\partial \dot{q}_{\text{ind}}^w} \right|_{q_{\text{ref}}} (\dot{q}_{\text{ind}}^w - \dot{q}_{\text{ref}}) + \left. \frac{\partial f}{\partial q_{\text{ind}}^w} \right|_{q_{\text{ref}}} (q_{\text{ind}}^w - q_{\text{ref}}) = 0
\] (15)

where the linearized mass, damping and stiffness matrices can be respectively identified as

\[
\left. \frac{\partial f}{\partial \ddot{q}_{\text{ind}}^w} \right|_{q_{\text{ref}}} = M_{\text{lin}}, \quad \left. \frac{\partial f}{\partial \dot{q}_{\text{ind}}^w} \right|_{q_{\text{ref}}} = C_{\text{lin}}, \quad \left. \frac{\partial f}{\partial q_{\text{ind}}^w} \right|_{q_{\text{ref}}} = K_{\text{lin}}
\] (16)

In this work, a partial linearization of the equations of motion is done and therefore, not the whole Eq. (14) is linearized. Due to the high non-linear behaviour of the contact forces, this term is not linearized while the suspension, inertia and quadratic terms in velocity forces are linearized. Consequently, the partially linearized equations of motion yield:

\[
M_{\text{lin}} (\ddot{q}_{\text{ind}}^w - \ddot{q}_{\text{ref}}) + C_{\text{lin}} (\dot{q}_{\text{ind}}^w - \dot{q}_{\text{ref}}) + K_{\text{lin}} (q_{\text{ind}}^w - q_{\text{ref}}) = B^T Q_{\text{contact}}
\] (17)

The linearized stiffness, damping and mass matrices are numerically calculated using the finite difference method with central differences where the function vector \( f \) of Eq. (15) is evaluated in the equilibrium position, velocity and acceleration respectively. Note that vector \( f \), no longer accounts for the contact forces \( Q_{\text{contact}} \)

4.3 Numerical interpolation in the linearized equations of motion

The partially linearized equations of motion presented in the previous section can accurately reproduce the dynamic behaviour of a railroad vehicle if the vehicle movement is in the proximity of the equilibrium positions where the mass, damping and stiffness matrices were calculated. In order to account for precise results when a railroad vehicle is running on an arbitrary track, a preprocessing stage must be performed. This preprocessing stage is described below:
• The different track streches where the vehicle can reach stable movement are identified and their corresponding equilibrium positions are evaluated.
• Linearized mass, damping and stiffness matrices are evaluated according to the different equilibrium positions.
• A linear interpolation between the sequentially linearized mass, damping and stiffness matrices is carried out in order to generate an adequate number of matrices to be used when a vehicle body is running on transition streches.

Once this preprocessing stage is done and during the dynamic simulation of a railroad vehicle, each vehicle body identifies what are the linearised matrices according to its track position.

5 CASE STUDY
In this section, a wheelset extracted from the ML95 trainset of the Lisbon subway company presented in [5] and running on an tangent and curved track without irregularities is presented.

5.1 Single wheelset case study
A single wheelset is studied running at a constant forward velocity $V = 10m/s$ on a tangent and curved ideal track. Its lateral position is misaligned $1mm$ respect to the equilibrium position. The mechanical properties of the wheelset as presented in [5] together with the track definition are listed in Table 1 where $\kappa_i$ is the curvature of strech $i$, $R_{\text{wheel}}$ the wheel radius and superscript wh refers to wheelset:

<table>
<thead>
<tr>
<th>Wheelset properties</th>
<th>ML95</th>
</tr>
</thead>
<tbody>
<tr>
<td>$m_{\text{wh}}$</td>
<td>933kg</td>
</tr>
<tr>
<td>$R_{\text{wheel}}$</td>
<td>0.43m</td>
</tr>
<tr>
<td>$I_{\text{xx}}^{\text{wh}}$</td>
<td>461.4kg · $m^2$</td>
</tr>
<tr>
<td>$I_{\text{yy}}^{\text{wh}}$</td>
<td>61.6kg · $m^2$</td>
</tr>
<tr>
<td>$I_{\text{zz}}^{\text{wh}}$</td>
<td>461.4kg · $m^2$</td>
</tr>
</tbody>
</table>

Table 1: Wheelset properties and track definition.

<table>
<thead>
<tr>
<th>Track definition - wheelset case study</th>
</tr>
</thead>
<tbody>
<tr>
<td>$s$</td>
</tr>
<tr>
<td>$s$</td>
</tr>
<tr>
<td>$s$</td>
</tr>
<tr>
<td>$s$</td>
</tr>
<tr>
<td>$s$</td>
</tr>
</tbody>
</table>

The purpose of this study is to evaluate the influence of the linearization when the wheelset goes on different streches of the track and compares the results to those obtained with the fully non-linear formulation proposed.

6 NUMERICAL RESULTS AND DISCUSSION
In this section, the numerical results of the case study proposed in Section 5 is discussed.

6.1 Single wheelset case study
The lateral movement of the single wheelset with the partially linearized and non-linearized formulation is presented on the left figure of Fig. (3) while the track centerline is on the right. Note that the y-axis on the track centerline figure of Fig. (3) is scaled in such a way that the track curvature can be observed. The lateral oscillations are in accordance to the unstable movement that the wheelset presents when running at a forward velocity of $V = 10m/s$ as in [1] and [2].
In the left figure of Fig. (3), one can see that the results obtained with the partially linearized model (red line) and the non-linearized one (blue line) are similar. Only a minor lateral amplitude difference can be observed. The wheelset is oscillating around its equilibrium position during the first track stretch. Then, when the wheelset reaches the trajectory coordinate $s = 50m$, it starts negotiating the transition stretch where it oscillates around the equilibrium position at the curve. In this stretch the equilibrium position implies a wheelset lateral displacement of $y^w = -3.9mm$. Later, at $s = 150m$ the wheelset negotiates a new transition stretch that leads to a tangent track at $s = 200m$. Note that the amplitude of the oscillations is increasing in time due to the system unstability.

In addition, a simple verification of the results has been done. If a wheelset with conical profiles is displaced in the lateral direction, a restoring force interacts and a periodical movement occurs. This movement is known as the Klingel movement [6] which states that the oscillation frequency $\omega_n$ can be expressed as $\omega_n = V \sqrt{2\gamma/(R \cdot b)}$, where $\gamma$ is the thread equivalent conicity, $R$ is the wheel radius and $b$ is the track gauge. In the proposed case study, the Klingel frequency has a value of $\omega_n = 0.62Hz$ that coincides with the simulation results of $\omega = 0.60Hz$. This slight difference can be explained due to the fact that the Klingel movement is purely a kinematic movement where no tangential contact forces are considered.

Computational efficiency is also compared. However, the fact to properly account for computational effort is a dedicated task that depends on a combination of hardware and software together with a proper code implementation. In this work, the computational effort is just compared for information purposes. The models are simulated on a Matlab® environment using object oriented programming concepts. A standard multi-core PC (without parallel programming) is used, and the computational time for the 20s simulation of the non-linearized and partially linearized wheelset models are 19.02s and 13.36s respectively, which means a 29.76% less time respect to the non-linearized model.

7 CONCLUSION

A procedure for linearize partially the multibody equations of motion of railroad vehicles has been presented. These equations of motion are first obtained as a transformation of the Newton-Euler equations of the vehicle bodies in terms of their independent coordinates, to then, numerically be
linearized respect to different equilibrium positions as Taylor series. According to the track geometry, different mass, damping and stiffness linearized matrices are calculated in order to interpolate them between the transition stretches of the track. Wheel-rail contact forces are not linearized but precalculated look-up tables that can take into account track irregularities are used to speed up the dynamic simulations. Results show good agreement between the non-linearized and partially linearized models. The computational cost saved with the procedure proposed can reach the 30% in a standard PC with Matlab® object oriented programming concepts used.

References
A Constraint Embedding Approach for Complex Vehicle Suspension Dynamics

Abhinandan Jain∗, Calvin Kuo#, Paramsothy Jayakumar†, Jonathan Cameron∗

∗Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive, Pasadena, California 91016, USA
jain@jpl.nasa.gov, jmc@jpl.nasa.gov

#Department of Mechanical Engineering
Stanford University
Palo Alto, California 91016, USA
calvink@stanford.edu

†U.S. Army TARDEC
6501 E 11 Mile Rd
Warren, MI 48397, USA
paramsothy.jayakumar.civ@mail.mil

Abstract
The goal of this research is to achieve close to real-time dynamics performance for allowing the closed-loop testing of unmanned ground vehicles (UGV) for urban as well as off-road scenarios. The overall vehicle dynamics performance is governed by the multibody dynamics model for the vehicle, the wheel/terrain interaction dynamics and the onboard control system. The topic of this paper is the development of computationally efficient and accurate dynamics model for ground vehicles with complex suspension dynamics. A challenge is that typical vehicle suspensions involve closed-chain loops which require expensive DAE integration techniques. In this paper, We illustrate the use the alternative constraint embedding technique to reduce the cost and improve the accuracy of the dynamics model for the vehicle.

1 INTRODUCTION
In this paper, we describe the constraint embedding approach for the modeling the dynamics of a 4-wheeled HMMWV vehicle, that has a double wishbone suspension and an associated spring-damper unit at each wheel. Each of these wheel suspensions contains a number of articulated bodies with multiple kinematic closed loops. Despite the large number of internal degrees of freedom, due to the constraints each suspension unit has only a single effective degree of freedom.

The standard approach for modeling closed-chain system dynamics [1] entails decomposing the system into a tree-topology system (or even a collection of independent bodies) and appending the closed-chain bilateral constraints to the equations of motion. A drawback of this approach is the increased computation for solving the equations of motion. Another serious drawback is the error drift that arises during the integration of the multibody dynamics equations of motion. This error drift is usually handled by the use of a differential-algebraic equation (DAE) solver and error correction algorithms to manage the constraint error over time, adding even more computational cost and accuracy error to the dynamics solution. Our desire for real-time performance require us to address these major computational drawbacks of the conventional approaches for closed-chain dynamics.

The recently developed constrained embedding (CE) method [2, 3] overcomes these drawbacks for closed-chain dynamics models. In this paper we describe the application of the constraint
embedding approach for the vehicle and suspension dynamics. The constraint embedded technique converts all constraint loops into compound bodies with variable configuration that have the same number of degrees of freedom as the number of independent degrees of freedom for the loops they replace. These compound bodies internally handle their internal degrees of freedom and constraints, effectively hiding them from the dynamics solver. The resulting system topology is once again a tree with only inter-body hinges and no bilateral constraints. The benefit of this approach is that structure-based $O(N)$ tree algorithms can be directly used to solve the dynamics, and this formulation results in an ODE instead of a DAE. Thus extra error control techniques are not needed. This method however is more complex to implement, since the aggregated bodies now have configuration dependent geometry. While CE method shares the minimal coordinates attribute with projection dynamics techniques [1, 4], its advantage lies in the preservation of the system’s tree topology that is necessary for the use of the low-cost structure-based tree algorithms.

In this paper we describe the CE modeling approach for the individual wheel suspensions, the overall vehicle dynamics model, and the adaptation of the recursive $O(N)$ dynamics algorithm for efficiently solving the equations of motion. While generic iterative methods can be used to solve the kinematics for the loops, we also describe analytical techniques that significantly improve performance speed up and accuracy. In this paper we focus only on vehicle dynamics. See references ([5] and [6] papers) for terramechanics and closed-loop shared control scenario modeling using this approach.

We begin in Section 2 with an overview of the $O(N)$ ODE techniques for solving the dynamics of a tree-topology dynamics system. Section 3 takes up the dynamics of non-tree topology systems, i.e. systems with closed-loop constraints that is typical of vehicle suspension systems. We provide an overview of the constraint embedding technique that solves closed-chain dynamics using $O(N)$ ODE techniques, and describe the key differences in handling aggregated bodies. Section 4 describes the HMMWV vehicle and the dynamics model including the CE model for its double-wishbone wheel suspensions. Finally, in Section 5 we describe analytical techniques that can be used for the double-wishbone suspension kinematics to further speed up the dynamics computations and improve their accuracy.

## 2 RECURSIVE TREE SYSTEM DYNAMICS

The equations of motion for a multibody system with tree topology (i.e. no closed loop constraints) and a $N$ degrees of freedom are of the form

$$\mathbf{T} = \mathbf{M}(\mathbf{\theta})\ddot{\mathbf{\theta}} + \mathbf{C}(\mathbf{\theta}, \dot{\mathbf{\theta}})$$ (1)

Here $\mathbf{M} \in \mathbb{R}^{N \times N}$ denotes the mass matrix for the serial-chain system, and $\mathbf{C} \in \mathbb{R}^N$ is the vector of velocity dependent nonlinear Coriolis and velocity dependent terms, and gravitational and external forces. The $N$ dimensional stacked vectors $\mathbf{\theta}$, $\dot{\mathbf{\theta}}$ and $\mathbf{T}$ denote the system generalized coordinates, generalized velocities and generalized forces. In this form, the tree-topology equations of motion can be propagated using an ODE integrator.

Using spatial operator techniques [3, 7], the following Newton-Euler Factorization expression for the mass matrix in Eq. 1 and $\mathbf{C}$ can be obtained:

$$\mathbf{M}(\mathbf{\theta}) = \mathbf{H}\phi\mathbf{M}\phi^*\mathbf{H}^* \in \mathbb{R}^{N \times N} \quad \text{and} \quad \mathbf{C}(\mathbf{\theta}, \dot{\mathbf{\theta}}) \overset{\Delta}{=} \mathbf{H}\phi(\mathbf{M}\phi^*\mathbf{a} + \mathbf{b}) \in \mathbb{R}^N$$ (2)

With $n$ denoting the number of bodies in the system, the $\mathbf{H} \in \mathbb{R}^{N \times 6n}$ and $\mathbf{M} \in \mathbb{R}^{6n \times 6n}$ spatial operators are block diagonal with the hinge axes and body spatial inertia matrices for each of the bodies being the diagonal elements respectively. The block lower-triangular $\phi \in \mathbb{R}^{6n \times 6n}$ operator's elements are the $6 \times 6$ rigid body transformation matrices for body pairs in the system. The $\mathbf{a}$ stacked vector contains the the body Coriolis accelerations, while $\mathbf{b}$ contains the body gyroscopic, external and gravitational forces for the system.
Further use of spatial operator techniques [3, 7] can be used to obtain the following analytical
Innovations factorization and inversion expressions for the mass matrix:

\[ M = H \phi M^* H^* \]
\[ M = [I + H \phi \mathcal{K}] D [I + H \phi \mathcal{K}]^* \]
\[ [I + H \phi \mathcal{K}]^{-1} = [I - H \psi \mathcal{K}] \]
\[ M^{-1} = [I - H \psi \mathcal{K}]^* D^{-1} [I - H \psi \mathcal{K}] \]

(3)

The component elements of the \( \psi, \mathcal{D}, \) and \( \mathcal{K} \) spatial operators are obtained from the following
tip-to-base articulated body (AB) Riccati equation recursion described here for the \( k \)th body:

\[ P^+(c) = \tau(c) P(c) \]
\[ P(k) = \sum_{c \in \mathcal{E}(k)} \phi(k, c) P^+(c) \phi^*(k, c) + M(k) \]
\[ \mathcal{D}(k) = H(k) P(k) H^* \]
\[ \mathcal{G}(k) = P(k) H^* \mathcal{D}^{-1}(k) \]
\[ \tau(k) = \mathcal{G}(k) H(k) \]

(4)

In the above, \( \mathcal{E}(k) \) denotes the set of bodies that are the immediate children of the \( k \)th body.

The analytical expression for \( M^{-1} \) in Eq. 3 allows to explicitly solve Eq. 1 explicitly and develop
the following expression for the generalized accelerations:

\[ \ddot{\mathbf{q}} = [I - H \psi \mathcal{K}]^* D^{-1} \left[ \mathcal{J} - H \psi (\mathcal{K} \mathcal{J} + P a + b) \right] - \mathcal{K}^* \psi^* a \]

(5)

Eq. 5 can be converted into the \( O(N) \) AB recursive forward dynamics algorithm. The tip-to-base
gather recursion steps for the \( k \)th body has the following form:

\[ \gamma^+(c) = \gamma(c) + \mathcal{G}(c) e(c) \]
\[ \gamma(k) = \sum_{c \in \mathcal{E}(k)} \phi(k, c) \gamma^+(c) + b(k) + P(k) a(k) \]
\[ e(k) = \mathcal{G}(k) - H(k) \gamma(k) \]
\[ \nu(k) = \mathcal{D}^{-1}(k) e(k) \]

(6)

The base-to-tip steps from body \( p \) to it’s child body \( k \) are as follows:

\[ \alpha^+(k) = \phi^*(p, k) \alpha(p) \]
\[ \ddot{\gamma}(k) = \nu(k) - \mathcal{G}^*(k) \alpha^+(k) \]
\[ \alpha(k) = \alpha^+(k) + H^*(k) \ddot{\gamma}(k) + a(k) \]

(7)

The AB algorithm is the lowest order algorithm available for solving the forward dynamics of

3 CLOSED CHAIN DYNAMICS

Multibody systems with closed-loop constraints can be decomposed into an tree-topology system
subject to explicit bilateral constraints. The decomposition is not unique. A common option is to
define the tree-topology system as consisting of all the component bodies as independent bodies
with the set of constraints containing all the loop constraints as well as constraints for all the inter-
body hinges as shown in Figure 1(a). In this fully-augmented (FA) approach the dynamics model
and constraints dimension is large but with sparse structure. In the alternative tree-augmented (TA) approach, the tree-topology system is chosen such that the number of explicit constraints
In the fully augmented model (a), all bodies are treated as independent bodies with inter-body constraints. In the tree augmented model (b), the system is decomposed into a tree system together with a minimal set of inter-body constraints. In the constraint embedding model (c), internal loops are aggregated into bodies to convert the system into a tree topology system.

is the minimum number as illustrated in Figure 1(b). The size of the dynamics model is much smaller, but the mass matrix has much less sparsity. In either case, the equations of motion for multibody systems with closed-loop constraints have the following form:

\[
\begin{bmatrix}
M & G_c^\top \\
G_c & 0
\end{bmatrix}
\begin{bmatrix}
\ddot{\theta} \\
-\lambda
\end{bmatrix}
= \begin{bmatrix}
\mathcal{J} - G_c \\
\dot{\mathcal{J}}
\end{bmatrix}
\]  

where \( \dot{\mathcal{U}} \triangleq \dot{\mathcal{U}}(t) - \hat{G}_c \dot{\theta} \in \mathbb{R}^{nc} \) (8)

Here \( G_c \) denotes the constraint matrix, and \( \lambda \) the Lagrange multipliers corresponding to the constraints. DAE integration techniques are required for solving the Eq. 8 dynamics model. One approach to solving the closed-chain dynamics equations of motion is to assemble the matrix on the left and the vector on the right in Eq. 8 and solve the linear matrix equation for the \( \ddot{\theta} \) generalized accelerations. This is especially attractive for the FA model, since the \( M \) matrix for this case is block diagonal and constant. Indeed, the whole matrix is highly sparse for this case. This approach is analyzed in detail in reference [4]. In the TA approach, the \( O(N) \) AB algorithm can be used for the tree-topology component and is described in reference [8].

3.1 Constraint Embedding Approach

At the heart of the constraint embedding strategy for closed-chain systems is the transformation of a non-tree topology system into a tree topology system. The approach is to isolate non-tree sub-graphs and remove them using aggregation to transform the system digraph into a tree. The constraint embedding transformation is illustrated illustrated in Figure 1(c). The constraint embedding strategy involves the following steps:

1. Decompose the non-tree digraph for the system into a spanning tree, \( \mathcal{T} \), and a collection of cut-edges for the constraints. The set of cut-edges is usually not unique.

2. For each cut-edge, identify the aggregation sub-graph, \( S \), for the sub-graph consisting of the node pair for the cut-edge.

   A procedure for creating this \( S \) aggregation sub-graph is as follows:

   (a) Identify the smallest sub-tree that contains the nodes in the cut-edge.

   (b) Remove the root node from this sub-tree to obtain the aggregation sub-graph \( S \) for the aggregated body.

At the conclusion of the constrained embedding process, all of the constraints are absorbed into the aggregated links. As a result, we once again have a tree-topology system and the mass matrix
factorization and inversion results, as well as the $O(N)$ AB algorithm for solving the dynamics as described in Section 2 can be extended and applied to the closed-chain system as discussed in detail in [2, 3].

3.2 Recursive CE Forward Dynamics

In this section we focus on the differences in the AB recursive dynamics procedure that are specific to the aggregated body $\mathcal{S}$. Towards this, let $S(\mathcal{S})$ denote the set of articulated rigid bodies contained within the $\mathcal{S}$ body, $n(\mathcal{S})$ the number of these bodies, $N_I(\mathcal{S})$ the number of generalized velocities associated with all the bodies in $S(\mathcal{S})$, and $N(\mathcal{S})$ the number of independent generalized velocities for the $\mathcal{S}$ body. The $\hat{\theta}(\mathcal{S})$ generalized velocities for the $\mathcal{S}$ body are the $N(\mathcal{S})$ independent generalized velocities among the sum total of $N_I(\mathcal{S})$ generalized velocities $\hat{\theta}_{\mathcal{S}}$ for the individual rigid bodies within $S(\mathcal{S})$. Let $X_{\mathcal{S}} \in \mathbb{R}^{N_I(\mathcal{S}) \times N(\mathcal{S})}$ denote the configuration dependent matrix that maps the $\hat{\theta}(\mathcal{S})$ independent generalized velocities into the $\hat{\theta}_{\mathcal{S}}$ set of internal generalized velocities such that

$$
\hat{\theta}_{\mathcal{S}} \triangleq \begin{bmatrix} \hat{\theta}(j_1) \\
\vdots \\
\hat{\theta}(j_{n(\mathcal{S})}) \end{bmatrix} = X_{\mathcal{S}} \hat{\theta}(\mathcal{S}) \quad \text{and} \quad \mathcal{T}(\mathcal{S}) = X_{\mathcal{S}}^* \begin{bmatrix} \mathcal{T}(j_1) \\
\vdots \\
\mathcal{T}(j_{n(\mathcal{S})}) \end{bmatrix} \quad \text{where} \ j_i \in S(\mathcal{S})
$$

(9)

Also, for the aggregated body

$$
M(\mathcal{S}) \triangleq \text{diag}\left\{ M(i) \right\}_{i \in S(\mathcal{S})} \in \mathbb{R}^{6n(\mathcal{S}) \times 6n(\mathcal{S})}
$$

$$
H(\mathcal{S}) \triangleq X_{\mathcal{S}}^* H_{\mathcal{S}} \in \mathbb{R}^{N(\mathcal{S}) \times 6n(\mathcal{S})} \quad \text{where} \ H_{\mathcal{S}} \triangleq \text{diag}\left\{ H(i) \right\}_{i \in S(\mathcal{S})} \in \mathbb{R}^{N_I(\mathcal{S}) \times 6n(\mathcal{S})}
$$

$$
\phi(\mathcal{S}, c) \triangleq \begin{bmatrix} \phi(p(c)) \\
\vdots \\
0 \end{bmatrix} \in \mathbb{R}^{6n(\mathcal{S}) \times 6} \quad \text{where} \ c \in S(\mathcal{S})
$$

$$
\phi(p, \mathcal{S}) \triangleq [\phi(p, j_1), \ldots, \phi(p, j_{n(\mathcal{S})})] \in \mathbb{R}^{6 \times 6n(\mathcal{S})} \quad \text{where} \ j_i \in S(\mathcal{S})
$$

(10)

$p(c)$ denotes parent body for the $c$ body, and $p$ is the parent body for the aggregated body. One noticeable difference is that the quantities associated with the $\mathcal{S}$ aggregated body have row/column dimension $6n(\mathcal{S})$ instead of just 6 encountered for regular rigid bodies. For the vectorial quantities we have

$$
b(\mathcal{S}) \triangleq \begin{bmatrix} b(j_1) \\
b(j_{n(\mathcal{S})}) \end{bmatrix} \in \mathbb{R}^{6n(\mathcal{S})} \quad \text{and} \quad a(\mathcal{S}) \triangleq \begin{bmatrix} a(j_1) \\
a(j_{n(\mathcal{S})}) \end{bmatrix} + H_{\mathcal{S}}^* X_{\mathcal{S}} \hat{\theta} \in \mathbb{R}^{6n(\mathcal{S})} \quad \text{where} \ j_i \in S(\mathcal{S})
$$

(11)

Though the dimensions are larger, we can also see that many of these quantities have highly sparse structure that can be used to reduce the cost of the steps in the AB forward dynamics recursions. The most computationally expensive part of the AB steps is the computation and inversion of the $D(\mathcal{S})$ symmetric, positive definite matrix in Eq. 4. Its size is the number of independent degrees of freedom for the aggregated link. Thus, the computational cost of the AB algorithm is no longer linear in the number of independent degrees of freedom for the aggregated links, but, instead, is (in the worst case) quadratic in the total degrees of freedom in the $\mathcal{S}$ aggregation sub-graph, and cubic in the number of independent degrees of freedom in the $\mathcal{S}$ sub-graph. These additional costs, however, are modest when the loops are of moderate size.
3.3 CE Kinematics

The CE dynamics solution process also requires the following kinematics computations:

1. forward kinematics computation that maps the independent \( \dot{\theta}(S) \) coordinates into the \( \dot{\theta}_S \) full generalized coordinates values for the \( S \) CE graph.
2. velocity kinematics computation of the \( X_S \) matrix in Eq. 9 that maps the \( \dot{\theta}(S) \) independent generalized velocities into the \( \dot{\theta}_S \) internal generalized velocities for the \( S \).
3. the \( X_S \dot{\theta}(S) \) vector needed in Eq. 11.

In this section we describe the general numerical procedures for carrying out the kinematics computations in the above steps, before exploring analytical techniques for the HMMWV suspension in Section 3.3 that are both faster and more accurate. The general method for carrying out the forward kinematics computation in step (1) is to use a Newton-like iterative procedure to converge on the solution that satisfies the constraints within the \( S \) subgraph.

We now derive the general expression for the configuration dependent \( X_S \) matrix for step (2). For loop constraints, we have an algebraic constraint on the relative velocities of a pair of physical closure nodes in the sub-graph. Denoting a representative pair of closure nodes as \( o \) and \( p \), such a constraint can be expressed as

\[
\delta_S = A(\gamma_o - \gamma_p) = A(j_o - j_p) \hat{\theta}_S = \gamma \hat{\theta}_S = \begin{bmatrix} \gamma_o \gamma_p \end{bmatrix}
\]

(12)

Here, \( A \) denotes the constraint on the relative spatial velocities between this pair of closure nodes, and \( j_o, j_p \) denote sub-graph Jacobians relating the generalized velocities of the sub-graph to the spatial velocities at the \( o \) and \( p \) closure nodes. Also, \( \gamma = A(j_o - j_p), \hat{\theta}_u \hat{\theta}_S \) is the complement of the \( \dot{\theta}(S) \) sub-vector in \( \hat{\theta}_S \) and represents the dependent generalized velocity coordinates. \( j_o, j_p \), etc., and \( y_1 \) and \( y_2 \) represent sub-blocks within \( Y \). When the \( \hat{\theta}_S \) generalized velocities satisfy the constraints within \( S \), the constraint velocity error \( \delta_S = 0 \). With partitioning chosen such that \( Y_1 \) is square and full rank it follows from Eq. 12 that:

\[
\hat{\theta}_u = \gamma^{-1}_1 y_2 \quad \Rightarrow \quad X_S = \begin{bmatrix} \gamma^{-1}_1 y_2 \\ I \end{bmatrix}
\]

(13)

The following derives an expression for \( \dot{X}_S \dot{\theta}(S) \) needed in step (3). With \( Z = \gamma^{-1}_1 y_2 \), we have

\[
\frac{dZ}{dt} = \gamma^{-1}_1 \frac{dy_2}{dt} y_2 + \gamma^{-1}_1 \frac{dy_1}{dt} y_1 - \gamma^{-1}_1 \frac{dy_1}{dt} y_2 + \gamma^{-1}_1 \frac{dy_2}{dt} y_1 - \gamma^{-1}_1 \frac{dy_2}{dt} y_2 = y_1 \frac{dy_1}{dt} - \gamma^{-1}_1 \frac{dy_1}{dt} y_2
\]

Thus,

\[
\dot{X}_S = \begin{bmatrix} \gamma^{-1}_1 \dot{y}_2 \\ 0 \end{bmatrix} = \begin{bmatrix} \gamma^{-1}_1 [\dot{y}_1 Z - \dot{y}_2] \\ 0 \end{bmatrix} = \begin{bmatrix} -\gamma^{-1}_1 \dot{y}_1 X_S \\ 0 \end{bmatrix} \quad \Rightarrow \quad \dot{X}_S \dot{\theta}(S) = \begin{bmatrix} -\gamma^{-1}_1 \dot{y}_1 \dot{\theta}_S \\ 0 \end{bmatrix}
\]

Now from Eq. 12 it follows that

\[
\delta_S = \gamma \hat{\theta}_S + \gamma \hat{\theta}_S \quad \Rightarrow \quad \gamma \hat{\theta}_S = [\delta_S] \hat{\theta}_S = 0 \quad \Rightarrow \quad X_S \dot{\theta}(S) = \begin{bmatrix} -\gamma^{-1}_1 \delta_S \hat{\theta}_S \\ 0 \end{bmatrix}
\]

(14)

Note that while \( \delta_S \) represents the velocity level constraint violation error within \( S \), \( \hat{\delta}_S \) represents the acceleration level constraint violation error. These quantities are easily computed using the normal kinematics procedures for given \( \hat{\theta}_S \) and \( \hat{\theta}_S \) values. \( [\delta_S] \hat{\theta}_S = 0 \) represents the acceleration level error with \( \hat{\theta}_S = 0 \), i.e. just the velocity dependent contribution to the acceleration level error. Eq. 13 and Eq. 14 together provide general solutions for computing \( X_S \) and \( \dot{X}_S \dot{\theta}(S) \) needed for solving the CE dynamics.
4 THE HMMWV VEHICLE

The HMMWV vehicle has four independently suspended wheels and a steering linkage controlling the front wheel pair as shown in Figure 2. The HMMWV employs a double wishbone suspension for all of its wheels that offer more robustness than the similar McPherson strut and less complexity than the multi-link suspension. Each of the wheel suspensions are connected to the vehicle chassis via bushings. Including the compliance of the bushings in the dynamics model leads to a tree topology model for the vehicle and its suspensions, and thus the efficient $O(N)$ AB dynamics algorithm and the ODE formulation from Section 2 can be directly applied for solving the vehicle’s equations of motion. An alternative modeling option for allowing larger integrator time steps is to treat the bushings as being infinitely stiff - and hence as forming bilateral constraints between the suspensions and the chassis. In this case, the vehicle dynamics topology with the double wishbone suspension and the steering linkage contains closed loops. Due to these loops, the vehicle model topology no longer has a tree topology, and the $O(N)$ AB algorithm and ODE formulation for tree systems can no longer be used. In the rest of this paper we describe the use of the constraint embedding technique for recovering the use of the $O(N)$ AB algorithms and ODE formulation for this more challenging vehicle dynamics model.

The double wishbone suspension (Figure 3) is comprised of two control arms that form a fourbar linkage with the spindle and the HMMWV chassis. In addition, the lower control arm forms a slider-crank linkage with the shock absorber and the HMMWV chassis. Finally, there is a tie-rod that connects to spindle forming the final closed chain in the suspension. The spindle loop is unique in that it falls outside the plane of the other two loops and its main purpose is to control the steering angle of the wheels, which are connected to the spindle. In the front suspensions, the tie rod is attached to the steering bar while in the rear suspensions, the tie rod is attached to the HMMWV chassis, which fixes the steering angle of the rear wheels.

The joints that connect the two control arms with the spindle in the physical suspension are ball and socket joints and have full rotational degrees of freedom. This is useful for the physical suspension in which design imperfections and sudden shocks can bring the control arm loop out of plane. In the absence of such non-idealities, it is more efficient to represent these joints with a universal joint with one rotational degree of freedom along the spindle axis and the other in plane with the control arm loop. The joints that connect the tie rod to the spindle is also manufactured as a ball and socket joint in the physical HMMWV, however this introduces an uncontrolled rotational
degree of freedom in the model in which the tie rod can freely spin about its own axis. Again, joint connecting the tie rod and spindle can be modeled more simply as a universal joint.

The steering mechanism consists of a simple fourbar linkage (Figure 4) comprised of the steering link, the pitman arm, the idler arm, and the HMMWV chassis. In our model, the steering is controlled by assigning a prescribed motion to the pitman arm. The ends of the steering arm are connected to the two front suspension tie rods and the resulting motion in the steering arm turns the front wheels about their spindle axes.

4.1 CE Model for the HMMWV Vehicle

A schematic for the double wishbone linkage for kinematic analysis is shown in Figure 5. In the figure, AD is the lower control arm, DF the upright arm, FG the upper control arm. BI is the lower shock absorber arm, HJ the upper shock arm and HI the compression. EK denotes the spindle arm, and KL the tierod. The point L denotes the end of the tierod. For the rear wheels, L is attached to the vehicle chassis via a ball joint and its position is therefore fixed. For the pair of front wheels, the L points are attached to points on the steering arm that is a part of the Pitman steering mechanism as shown in Figure 6. Steering is accomplished by changing the steering angle, which causes the steering arm to move the tierods and change the wheel orientations.

Each wheel suspension has seven bodies (without including the wheel) and three constraints re-
sulting in a single degree of freedom for each suspension. Each suspension is decomposed into a
tree-topology system with the hinges at J, G and L being treated as constraints. Constraint embed-
dding is used to model each suspension system as a individual single degree of freedom aggregated
body. Each such aggregated body is attached to the chassis parent body and in turn has a single
wheel as a child body. The tierod end point locations for the front wheel suspensions are attached
to the Pitman steering mechanism’s steering link and are movable. While in principle the steering
mechanism introduces additional constraints between the front wheel suspensions, for the pur-
poses of this model we treat the steering mechanism kinetically so that its effect on the dynamics
is only to set the position of the tierod end points as a function of the steering wheel angle. Thus
the CE dynamics model consists of the chassis body, four suspension aggregated bodies and four
wheel bodies with overall fourteen degrees of freedom. The CE $O(N)$ method described in Sec-
tion 3 can be used to solve the equations of motion of this ODE model to simulate the vehicle
dynamics.

5 ANALYTICAL DOUBLE WISHBONE KINEMATICS

Section 3.3 describes a numerical approach for computing the $X_\Theta$ needed for the constraint embed-
dding dynamics. While the method is general, replacing it with analytical methods when possible
provides a way to improve computational speed and accuracy.

In this section we derive analytical expressions for the forward kinematics, as well as the ve-
locity level $X_\Theta$ for planar four-bar linkages, which will provide a stepping stone for developing
expressions for the full HMMWV wheel suspensions. Each suspension has only a single degree
of freedom, and we choose the generalized coordinate with the lower control arm, $\angle QAD$, as the
independent generalized coordinate and denote it by the symbol $\theta$.

The three loops in the suspension system are:

1. the ADFG lower/upper control arm loop consisting of the planar four-bar linkage containing
the upper and lower control arms;

2. the ABJ shock absorber loop involving the planar shock absorber mechanism;

3. the EKL spindle loop involves the non-planar spindle and tie-rod mechanism.
We now derive analytical expressions for the forward kinematics as well as the velocity kinematics for each of the loops.

5.1 Lower/Upper control arm kinematics

For the forward kinematics we need to determine the values of all the dependent angles for a value of the independent angle. We do the initial derivation using absolute angles and use this to obtain expressions for the relative angle generalized coordinates.

We use the following symbols for the four-bar parameters for the derivations within this section:

\[ a = |GA|, \quad b = |AD|, \quad c = |DF|, \quad d = |FG| \]
\[ \theta_3 = \angle PFO, \quad \theta_4 = \angle QGF \]

The forward kinematics problem for the lower/upper control arm loop is to determine the dependent generalized coordinates \( \angle NDO, \angle OFG \) and \( \angle RGA \) as functions of the \( \theta = \angle QAD \) independent coordinate.

For 2D kinematic analysis, we use a derivation based on complex numbers and the 2D exponential \( \exp(x) = \cos(x) + i\sin(x) \). We have

\[ a + b \exp(i\theta) = c \exp(i\theta_3) + d \exp(i\theta_4) \quad (15) \]

Equating the real and imaginary parts leads to

\[ a + b \cos(\theta) = c \cos(\theta_3) + d \cos(\theta_4) \quad (16) \]
\[ b \sin(\theta) = c \sin(\theta_3) + d \sin(\theta_4) \]

Thus

\[ c \cos(\theta_3) = a + b \cos(\theta) - d \cos(\theta_4) = x - d \cos(\theta_4) \quad \text{where} \quad x \triangleq a + b \cos(\theta) \quad (17) \]
\[ c \sin(\theta_3) = b \sin(\theta) - d \sin(\theta_4) = y - d \sin(\theta_4) \quad \text{where} \quad y \triangleq b \sin(\theta) \]

Summing up the squares of both sides leads to

\[ c^2 = x^2 + y^2 + d^2 - 2d(x \cos(\theta_4) + y \sin(\theta_4)) \]
\[ \Rightarrow x \cos(\theta_4) + y \sin(\theta_4) = \frac{x^2 + y^2 + d^2 - c^2}{2d} \quad (18) \]

Dividing both sides by \( \sqrt{x^2 + y^2} \) leads to

\[ \cos(\theta_4 - \gamma) = \frac{x^2 + y^2 + d^2 - c^2}{2d \sqrt{x^2 + y^2}} \quad \text{where} \quad \gamma \triangleq \tan^{-1}\left(\frac{y}{x}\right) \quad (19) \]

Thus

\[ \theta_4 = \gamma \pm \cos^{-1}\left(\frac{x^2 + y^2 + d^2 - c^2}{2d \sqrt{x^2 + y^2}}\right) \quad (20) \]

Note that we have two possible solutions for \( \theta_4 \). Then from Eq. 17

\[ \theta_3 = \tan^{-1}\left(\frac{y - d \sin(\theta_4)}{x - d \cos(\theta_4)}\right) \quad (21) \]

With the solution for the absolute angles, the values of the dependent generalized coordinates are

\[ \angle NDO = \pi - (\theta - \theta_3), \quad \angle OFG = \theta_4 - \theta_3 \quad \text{and} \quad \angle RGA = \pi - \theta_4 \quad (22) \]
The above provide the analytical forward kinematics expressions for the lower/upper control arm four-bar linkage. For the velocity level expressions, it follows from Eq. 22 that

\[ \dot{\angle NDO} = -(\dot{\theta} - \dot{\theta}_3), \quad \dot{\angle OFG} = \dot{\theta}_4 - \dot{\theta}_3 \quad \text{and} \quad \dot{\angle RGA} = -\dot{\theta}_4 \]  

\hspace{1cm} (23)

We thus need to determine analytical expressions for \( \dot{\theta}_3 \) and \( \dot{\theta}_4 \). Relationships that map the independent We begin by taking the time derivative of Eq. 15 to obtain

\[ \dot{\theta}_b \exp(i\theta) = \dot{\theta}_3 \exp(i\theta_3) + \dot{\theta}_4 \exp(i\theta_4) \]

\[ \Rightarrow \dot{\theta}_b \exp(i(\theta - \theta_4)) = \dot{\theta}_3 \exp(i(\theta_3 - \theta_4)) + \dot{\theta}_4 \exp(i\theta_4) \]  

\hspace{1cm} (24)

Equating the imaginary sides of both sides leads to

\[ \dot{\theta}_b \sin(\theta - \theta_4) = \dot{\theta}_3 \sin(\theta_3 - \theta_4) \]  

\[ \Rightarrow \dot{\theta}_3 = p\dot{\theta} \quad \text{where} \quad p = \frac{b \sin(\theta - \theta_4)}{c \sin(\theta_3 - \theta_4)} \]  

\hspace{1cm} (25)

Similarly

\[ \dot{\theta}_4 = q\dot{\theta} \quad \text{where} \quad q = \frac{b \sin(\theta - \theta_3)}{d \sin(\theta_4 - \theta_3)} \]  

\hspace{1cm} (26)

This leads to the following closed-form expression for the four-bar portion of the \( X_{\Theta} \):

\[ X_{\Theta} = \begin{bmatrix} p - 1 \\ q - p \\ -q \end{bmatrix} \]  

\hspace{1cm} (27)

These equations represent a complete set of analytical kinematic expressions needed for constraint embedding solution process for a four-bar linkage.

### 5.2 Shock absorber loop kinematics

We use the following symbols for the shock absorber loop parameters within this section:

\[ x = | AJ |, \quad y = | AB |, \quad z = | BJ | \]

\[ \beta = \angle CQJ, \quad \eta = \angle CQB, \quad \gamma = \angle MBJ \]

The forward kinematics problem here is to determine the dependent generalized coordinates \( \angle DBJ \) and the signed magnitude \( r \) of \( HI \) as functions of the \( \theta \) independent coordinate. The assumption is that \( H \) and \( I \) coincide when the shock absorber compression is zero. We have

\[ x \exp(i\beta) = y \exp(i\eta) + z \exp(i\gamma) \]

\[ \Rightarrow z \exp(i\gamma) = x \exp(i\beta) - y \exp(i\eta) \]  

\hspace{1cm} (28)

Note that \( x, \beta, | BI | \) and \( | HJ | \) are constant and do not change over time, and \( \eta = \theta - \pi/2 \). It follows from the real and imaginary parts of Eq. 28 that

\[ \gamma = \tan^{-1}\left(\frac{x \sin(\beta) - y \sin(\eta)}{x \cos(\beta) - y \cos(\eta)}\right) \quad \text{and} \quad z = \frac{x \cos(\beta) - y \cos(\eta)}{\cos(\gamma)} \]

Thus analytical expressions for the shock absorber loop’s generalized coordinates are thus

\[ \angle DBJ = \gamma - \eta \quad \text{and} \quad r = z - (| BI | + | HJ |) \]

For velocity kinematics, time differentiating Eq. 28 leads to

\[ 0 = i\eta y \exp(i\eta) + i\gamma z \exp(i\gamma) + \gamma \dot{z} \exp(i\gamma) \]  

\hspace{1cm} (29)
Multiplying both sides by \( \exp(-i\gamma) \) leads to

\[
0 = i\eta y \exp(i(\eta - \gamma)) + i\gamma z + \dot{z}
\]

Equating the real and imaginary parts results in

\[
0 = -\dot{\eta} y \sin(\eta - \gamma) + \dot{\gamma} z \\
\Rightarrow \quad \dot{\gamma} = -\frac{y \cos(\eta - \gamma)}{z} \eta
\]

Since \( \dot{\eta} = \dot{\theta} \) and \( \dot{\tau} = \dot{z} \), the expressions for the generalized velocities for the shock absorber loop are

\[
\angle DBJ = - \left( 1 + \frac{y \cos(\eta - \gamma)}{z} \right) \dot{\theta} \\
\dot{\tau} = y \sin(\eta - \gamma) \dot{\theta}
\]

The contributions to \( X_\Theta \) are:

\[
X_\Theta = \begin{bmatrix} - \left( 1 + \frac{y \cos(\eta - \gamma)}{z} \right) \\ y \sin(\eta - \gamma)/\tau \end{bmatrix}
\] (30)

Note that

5.3 Spindle/tierod loop kinematics

We use \( \alpha \) to denote the 1 degree of freedom generalized coordinate for the spindle’s rotation about the DF upright arm. The \( \mathcal{R}(\alpha) \) rotation matrix associated with this generalized coordinate has the form

\[
\mathcal{R}(\alpha) = \begin{pmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{pmatrix}
\] (31)

The tierod has 2 degree of freedom generalized coordinates for rotations about the X and Z successive axes at K. We denote these coordinate angles as \( \chi \) and \( \zeta \), respectively. The forward kinematics for the spindle/tierod look requires solving for \( \alpha, \chi \) and \( \zeta \) as a function of the \( \theta \) independent generalized coordinate. We have

\[
\overrightarrow{EL} - \overrightarrow{EK} = \overrightarrow{KL}
\]

Given a value for the independent coordinate \( \theta \), the location of O is known, and thus so is the vector \( \overrightarrow{EL} \). Thus

\[
|\overrightarrow{EL}|^2 + |\overrightarrow{EK}|^2 - 2(\overrightarrow{EL}^* \overrightarrow{EK}) = |\overrightarrow{KL}|^2
\]

Also \( \overrightarrow{EK} = \mathcal{R}(\alpha) \overrightarrow{EK}_0 \), where \( \overrightarrow{EK}_0 \) is the vector for the unrotated spindle arm. Thus

\[
\overrightarrow{EL}^* \mathcal{R}(\alpha) \overrightarrow{EK}_0 = \frac{1}{2} \left( |\overrightarrow{EL}|^2 + |\overrightarrow{EK}|^2 - |\overrightarrow{KL}|^2 \right)
\]

Let the elements of the vectors \( \overrightarrow{EL} \) and \( \overrightarrow{EK}_0 \) in the vertical arm frame be given by

\[
\begin{align*}
\overrightarrow{EL} &= \begin{bmatrix} \overrightarrow{EL}(x) \\ \overrightarrow{EL}(y) \\ \overrightarrow{EL}(z) \end{bmatrix} \\
\overrightarrow{EK}_0 &= \begin{bmatrix} \overrightarrow{EK}_0(x) \\ \overrightarrow{EK}_0(y) \\ \overrightarrow{EK}_0(z) \end{bmatrix}
\end{align*}
\]

With

\[
A \triangleq \overrightarrow{EL}(x) \overrightarrow{EK}_0(x) + \overrightarrow{EL}(y) \overrightarrow{EK}_0(y) \\
B \triangleq -\overrightarrow{EL}(x) \overrightarrow{EK}_0(y) + \overrightarrow{EL}(y) \overrightarrow{EK}_0(x)
\]
Eq. 31 leads to

\[ A \cos(\alpha) + B \sin(\alpha) = X \text{ where } X \triangleq \frac{1}{2} \left( |\vec{EL}|^2 + |\vec{EK}|^2 - |\vec{KL}|^2 \right) - \vec{E}K(z)\vec{E}K_0(z) \]

Defining

\[ \beta \triangleq \tan^{-1} \left( \frac{B}{A} \right) \]

we have

\[ \cos(\alpha - \beta) = \frac{X}{\sqrt{A^2 + B^2}} \Rightarrow \alpha = \beta \pm \cos^{-1} \left( \frac{X}{\sqrt{A^2 + B^2}} \right) \]

Once \( \alpha \) value is determined, the location of the tierod axes at \( K \) is known and so is the vector \( \vec{KL} \).

The \( \chi \) and \( \zeta \) tierod generalized coordinates are then simply the elevation and azimuth angles for the \( \vec{KL} \) vector as seen from the spindle’s frame. Using the elements of \( \vec{KL} \) in the spindle fixed frame we then have

\[ \chi = \sin^{-1} \left( \frac{\vec{KL}(z)}{|\vec{KL}|} \right) \text{ and } \zeta = \tan^{-1} \left( \frac{\vec{KL}(y)}{\vec{KL}(x)} \right) \]

For velocity kinematics, we need to solve for the \( \dot{\alpha}, \dot{\chi} \) and \( \dot{\zeta} \) generalized velocities for the spindle loop as a function of the \( \dot{\theta} \) independent generalized velocity. For a given, \( \dot{\theta} \), we can use the analytical velocity expressions derived so far in this expression to compute the \( v_E \) linear velocity of point \( E \) on the spindle with respect to \( A \) on the chassis. Since the end of the tierod \( L \) is constrained by a ball joint to the chassis, the relative motion from the spindle loop generalized velocities when combined with \( v_E \) has to result in zero linear velocity at \( L \). Thus

\[ v_E + \mathcal{J}_{EL} \begin{bmatrix} \dot{\alpha} \\ \dot{\chi} \\ \dot{\zeta} \end{bmatrix} = 0 \Rightarrow \begin{bmatrix} \dot{\alpha} \\ \dot{\chi} \\ \dot{\zeta} \end{bmatrix} = -\mathcal{J}_{EL}^{-1}v_E \]

where \( \mathcal{J}_{EL} \) denotes the \( 3 \times 3 \) Jacobian that maps the spindle generalized velocities into the linear velocity of \( L \) with respect to \( E \) on the spindle. The columns of this Jacobian are simply the cross product of the vector from the spindle hinge axis location to \( L \) and the hinge axis for each of the three hinge axes. Additionally, with \( \mathcal{J}_{AE} \) and \( \mathcal{J}_{DE} \) denoting the \( 3 \times 1 \) Jacobian matrices for the \( v_E \) linear velocity from the \( A \) and \( D \) hinge degrees of freedom, we have

\[ v_E = [\mathcal{J}_{AE}, \mathcal{J}_{DE}] \begin{bmatrix} \dot{\theta} \\ \angle \text{NDO} \end{bmatrix} \overset{27}{=} [\mathcal{J}_{AE} + (p-1)\mathcal{J}_{DE}]\dot{\theta} \]

\[ \Rightarrow \begin{bmatrix} \dot{\alpha} \\ \dot{\chi} \\ \dot{\zeta} \end{bmatrix} = -\mathcal{J}_{EL}^{-1}[\mathcal{J}_{AE} + (p-1)\mathcal{J}_{DE}]\dot{\theta} \]

Eq. 32 defines the contribution of the spindle loop to the \( X_{\dot{\theta}} \) matrix for the velocity kinematics.

5.4 Front steering kinematics

Figure 6 shows a schematic for the Pitman steering mechanism for the front wheels. With \( T \) and \( W \) being points on the chassis, the \( TU \) and \( WV \) links are the left and right idler arms respectively that connect to the \( LUVL \) steering link. The \( L \) end points of the steering link are connected to the tierods for the front wheels. \( TUVW \) represents a planar four-bar loop within the mechanism. Steering changes the \( \angle \text{XTU} \) angle causing the steering link, and consequently the left and right wheel tierods to move and change the orientation of the front wheels. For the purposes of dynamics
modeling, the only contribution of the steering mechanism is to the positioning of the tierod end points for the front wheels. The analytical approach for four-bar mechanism forward kinematics used in Section 5.1 can be used for the Pitman steering to determine its shape as a function of the steering angle and consequently the location of the tierod end points. The instantaneous location of the tierod endpoints are used in the spindle kinematics described in Section 5.3 for the front suspensions.

6 CONCLUSIONS

In this paper we describe in detail the application of constrained embedding technique for the modeling of the dynamics of a HMMWV vehicle with double wishbone suspension systems. Constraint embedding allows us to formulate the dynamics as an ODE system, and in a way that preserves the underlying structure of the system so that low-cost recursive methods for minimal coordinate systems can be applied for solving the equations of motion. We further illustrate analytical kinematic techniques for the HMMWV double wishbone suspension that can be used to speed up and improve the accuracy of the overall vehicle dynamics.

ACKNOWLEDGMENTS

The research described in this paper was performed at the U.S. Army TARDEC, and at the Jet Propulsion Laboratory (JPL), California Institute of Technology, under contract with the National Aeronautics and Space Administration.1

REFERENCES


[6] Jonathan M Cameron, Steven Myint, Calvin Kuo, Abhinandan Jain, Havard Grip, Paramsothy Jayakumar, and Jim Overholt. Real-Time and High-Fidelity Simulation Environment for Au-


A Dynamic-Based Approach for Road Vehicle Design:
Application to a Three-Wheeler

A. Verlé, P. Fisette

1 Center for Research in Energy and Mechatronics
Université catholique de Louvain (UCL)
Place du Levant 2, B-1348 Louvain-la-Neuve, Belgium
[aubain.verle, paul.fisette]@uclouvain.be

ABSTRACT
Nowadays, multibody simulations are widely exploited to validate and analyze mechanical systems whose functioning requires high dynamic performances. A global optimization of those dynamic systems can be carried out, producing more satisfactory results than classical approaches. We focus in this work on the family of road vehicles, characterized by a rather large number of design parameters and we propose a design methodology which takes dynamic issues into account from the earliest stages. The system parameters are classified in order to be processed afterwards by the suitable numerical module of the global design algorithm, rather than by the optimizer itself. The application of the proposed method concerns the optimization of the suspensions of a new concept of urban motorized three-wheeler under development in our laboratory.

Keywords: Optimization, Dynamic performances, Active tilting, Three-wheeler, Multibody.

1 INTRODUCTION
Nowadays, multibody simulations are widely exploited to validate and analyze mechanical systems whose functioning requires high dynamic performances. Thanks to the capabilities of the present computers and clusters, and given the maturity of the recent numerical algorithms, a global optimization of those dynamic systems can be carried out, producing—in a decent simulation time—more satisfactory results than classical approaches like parametric study, experimental plan or sub-systems optimization. In this respect, we focus in this work on the family of road vehicles, characterized by a rather large number of design parameters suited for optimization and we propose a design methodology which takes dynamic issues into account from the earliest stages. In the proposed approach, the system parameters are classified in order to be processed afterwards by the suitable numerical module of the global design algorithm, rather than by the optimizer itself. The application of the proposed method concerns the optimization of the suspensions of a new concept of urban motorized three-wheeler under development in our laboratory.

The proposed optimization method is part of a global design process which contains three successive steps.

Above all, it is assumed that the topology of the vehicle (i.e., type of joints, front and rear suspension linkages, etc) has been fixed by a preliminary mechanical design. This first step also provides realistic initial values for masses, inertia of the main vehicle components.

The second step is the core of the present work. Starting from the above initial design, the parameters are classified according to specific criteria according to their role in the design process (Section 2). Then, based on well-targeted simulations, the optimization process can actually start to determine the optimal set of parameters, such as component lengths, joints location and orientation, suspension stiffness, controller gain, etc. The objective function is clearly application-dependent and can represent, as usual, a weighted combination of several criteria such as the
Figure 1. The design of a tilting tricycle requires a dynamic-based engineering approach.

minimization of the dissipated power in the active suspension of the three-wheeler or of the front wheels load transfer in curve.

On the basis of the above optimized solution, the final mechanical design of the vehicle can start, with some possible iterations on the three preceding steps.

In Section 2, we describe the three-wheeler taken as example throughout the paper and, in particular, its tilting front wheel-axle units. Then, Section 3 gives an overview of the multibody formalism and the numerical methods used for dynamic simulation. We present the optimization methodology and discuss its interest in Section 4. Finally we give some relevant results in Section 5, before concluding.

2 VEHICLE DESCRIPTION

In our case, a reverse-trike configuration (i.e. two front wheels, one rear) has been adopted: the main justification involves braking deceleration capabilities as compared with the acceleration produced by rear motorization. The rear axle unit is the same type as a motorbike’s: a one-wheeled axle is far simpler than a tilting four-wheeler and tilts naturally.

The front wheel-axle unit is comparable to a traditional double wishbone except for the shocks’top connection, which is not directly linked to the chassis but to an intermediate component: an articulated T-shaped pendulum (Figure 2). This mounting provides the front axle unit with an additional degree of freedom (d.o.f.) corresponding to the tilting motion. Figure 2 illustrates each of the front axle d.o.f.: the top picture shows the reference position, the three others refer to the hop, steering and tilting motions respectively. The tilting actuator torque acts between the chassis and the T-pendulum. This parallel suspension topology, similar to existing Peugeot Metropolis front suspensions, is preferred over a serial one, such as that of the Piaggio MP3, because it allows us to design a less massive structure and limit suspension vibrations.

To describe the vehicle multibody model, 31 parameters are needed for the front axle unit and 12 for the rear, leading to a total of 43 parameters to be potentially optimized. The three-wheeler motion is described by 29 generalized coordinates subjected to 8 loop constraints, leading to 13 d.o.f. and 8 fixed coordinates.

Figure 2. Front axle unit.
The three-wheeler application appears to be very suited to the proposed methodology because it introduces numerous and varied difficulties:

- the front suspension is kinematically complex, with 3D loops of bodies;
- the vehicle is naturally unstable: The associated difficulties are multiple;
- the design must consequently include a reliable tilting controller;
- the steering behavior must be simulated by a driver controller;
- the expected performances are very different in nature;
- the mechanical design and specifications involve very diversified design constraints;
- the optimization parameters are also numerous and different in nature. Moreover, they are strongly interdependent in their relation with target performances.

Therefore, the development of a specific optimization scheme is fully justified.

3 GENERAL MULTIBODY ANALYSIS ENVIRONMENT

The goal of this section is to summarize the basic formalism and numerical methods used in the subsequent developments. The optimization methodology, described in Section 4, relies on multibody models whereby the vehicle morphology can be easily described. The Robotran generator [2] uses this description to provide symbolic C-function for the mass matrix $M$ and dynamic vector $c$, the constraints $h$ and the jacobian matrix of constraint $J$ to build the following system:

$$M(q)\ddot{q} + c(q, \dot{q}, frc, trq, g) = Q(q, \dot{q}) + J^T \lambda$$

$$h(q) = 0$$

$$\dot{h}(q, \dot{q}) = J(q)\dot{q} = 0$$

$$\ddot{h}(q, \dot{q}, \ddot{q}) = J(q)\ddot{q} + J\dot{q}(q, \dot{q}) = 0$$

in which $q$ are the joint generalized coordinates of the multibody system (mbs). The coordinate partitioning method [1] is used to select the best set of independent coordinates $q_u$ and dependent coordinates $q_v$. The number of degrees of freedom of the mbs is given by the number of $q_u$. After some matrix manipulations, the previous DAE system is reduced to an ODE system, in terms of the independent coordinates:

$$M_r\ddot{q}_u + F_r(\dot{q}_u, q_u) = 0$$

3.1 Three-wheeler additional models

Here, we describe the additional models required to simulate the three-wheeler. We then discuss the influence of the model quality on the results.

3.1.1 Tire model

The vehicle is equipped with scooter tires because it exhibits a similar dynamics. Thus, the simulator needs a tire model especially adapted to the scooter-moto type. The road-tire contact forces are computed according to the reduced static model of Pacejka [4] which excludes all asymmetric term. The parameters of these model have been identified by Sharp and al. [3]. A well-suited road-tire contact model is the key point to produce quality results. In fact, if a "rolling without slipping" constraint would be sufficient to study the stability of the three-wheeler, a more refined
model is needed to consider issues like under/over-steering. By the way, in the next future, we intend to complete the model with a relaxation length term to take the internal dynamic of tires into account.

### 3.1.2 Steering controller

During a given simulation, the steering column rotation, controlled in position, is set by two ways:

In case of a real time simulation, the steering orientation is provided by a game steering wheel connected to the C-code via the SDL\(^1\) library.

In case of a fast simulation, the information is computed by a dedicated controller. A predefined trajectory is described by a cubic spline. The rotation of the steering column \(\mu\) is given by a non-linear function of the trajectory curvature \(\rho\), a proportional-derivative term on the difference, \(\Delta\sigma\), between the current orientation of the vehicle \(\sigma_v\) and the curve orientation \(\sigma_t\) and a proportional-integral-derivative term on the difference, \(\varepsilon\), between the current position of the vehicle and the spline:

\[
\mu = f_{\rho\mu}(\rho) + PD(\Delta\sigma) + PID(\varepsilon)
\]  

The orientation and the curvature are evaluated at a target point on the trajectory, at a distance from the vehicle which depends on its speed.

![Figure 3. The steering control strategy.](image)

### 3.1.3 Tilt controller

As explained before, the tilt motion is controlled by an actuator that acts on the T-pendulum. At present, it is controlled in torque via a PID on the roll \(\phi\) angle of the three-wheeler:

\[
T_{act} = PID(\phi_{cmp} - \phi_{msr})
\]  

The computed roll \(\phi_{cmp}\) is based on three main current states: the speed, the steering wheel rotation and the rotational speed. The measured roll will be determined via an IMU\(^2\) in case of the real vehicle and is computed via a kinematic sensor in the simulation.

### 3.2 Equilibrium solution

The general equilibrium algorithm is depicted in Figure 4. An iterative solver \(S1\) based on the Newton-Raphson algorithm (NR) is used to solve the equilibrium equations:

\[
F_i(q_u) = 0
\]

\(^1\)Simple Directmedia Layer
\(^2\)Inertial Measurement Unit
3.3 Modal analysis

The $F_r$ vector must first be linearized with respect to $\dot{q}_u$ and $q_u$ around the equilibrium solution $q_u^{eq}$: we use an iterative parabolic fitting method. The system of Equation (5) is transformed into the linearized form:

$$M_r \Delta \ddot{q}_u + G_r \Delta \dot{q}_u + K_r \Delta q_u = 0$$  \hspace{1cm} (9)

Then, the Gsl$^3$ library is used to find, for each eigenmode: the eigenvalues, the normalized eigen vectors, the frequency $\omega$ and the damping rate $\xi$. For the three-wheeler study, a comparison algorithm is used to select modes related to front and rear suspension motion. The frequency and damping factor of these modes are considered as a good evaluation of comfort.

3.4 Time simulation

As depicted in Figure 5, the direct dynamics module —also called the simulator— is based on a time integration of both mechanical dynamics Equations (5) and user model state equations. The variables of this time integration are the independent coordinates $q_u$ and the user model states $z$. Up to now, a fixed time step Runge-Kutta method is used with a time step of $5e^{-4}$s, enough small to allow the multibody loop closure convergence with NR. Thanks to a recursive formulation in relative coordinates and to the algebraic simplifications achieved by the Robotran symbolic generator, each dynamic simulation (C language) only takes 9 s of computation time for 11 s of real time (run on a single core of an Intel® i7). By this way, real time simulation is also possible.

4 OPTIMIZATION METHODOLOGY

In this section, we present the optimization methodology. The goal of the method is the minimization of an objective function, characterizing the quality of the vehicle, via the evolution of predefined parameters. In this work, those parameters can come from 4 natures of data: a component of a geometrical vector, an orientation angle, a length associated with a distance constraint or any user-model parameter. There are some constraints associated with the problem. As depicted in Figure 6, the optimizer updates values to a subset of parameters, called *optimized parameters* $p_o$, while a "trial" evaluates the objective function corresponding to the vehicle performances.

$^3$GNU Scientific Library
The trial is divided into three main steps:

- **the data conditioner**: gives a value to non-optimized parameters which match the design constraints. It forms a multibody system with a complete and coherent data setting and ready to be run by the simulator.

- **the simulator**: tests the vehicle on a trajectory via a time simulation.

- **the evaluator**: computes the objective function for the current vehicle on the basis of the data arising from the simulator.

![Diagram](image)

**Figure 6.** The general optimization scheme.

### 4.1 Optimizer

Given the various natures of the parameters and the complex and non-linear relations between parameters and the objective function, it is more than likely that the problem is not convex as it stands. A way to find an optimal solution is to use a non-deterministic evolutionary-type algorithm like CMA-ES[5]. The algorithm uses a predetermined number of particles \(N_{\text{part}}\), which represent vehicle candidates. At each generation, the algorithm produces \(N_{\text{part}}\) vector parameters \(p_o\) and evaluates each of them via the trial. The optimized parameters of a generation are computed by the CMA-ES strategy based on the \(p_o\) vectors of the previous generation and the associated evaluations. Note that the optimizer doesn’t deal directly with the \(p_o\) parameters but with a normalized parameters vector \(x_o\) which follows the rule:

\[
x_o = \frac{p_o - p_{\text{omin}}}{p_{\text{omax}} - p_{\text{omin}}} \tag{10}
\]

where \(p_{\text{omin}}\) and \(p_{\text{omax}}\) are the range limit for \(p_o\), given by the user. So the optimizer only deals with \([0 \to 1]\) range parameters and all are treated equally despite of their nature, units and range. The process stops when one of the following convergence criteria is reached: the objective function variation, from a generation to another, is lower than a given threshold or the variation of the norm of the \(x_o\) vector norm is lower than another threshold.

An important setting of the optimizer is the number of particles by generation. On one hand, the required computer resources are directly proportional to this number, which must be minimized. On the other hand, if the number of particles is insufficient, the optimization may have difficulties to converge. By experience, we have found that \(N_{\text{part}} > 2N_{p_o}\) must be respected to ensure a good convergence.
4.2 Evaluator

As usual, the objective function must decrease as the "quality" of the vehicle increases. But, first of all, the vehicle must be mechanically possible, i.e. satisfy both design and loop constraints. As discussed in the following section, for a non-realistic vehicle, it is difficult to assess the gap with respect to a possible one. Despite the fact that we largely penalize it, if too many particles are in the case, the optimization can’t succeed.

The second goal of the three-wheeler is to reach the point of arrival of the test trajectory. If it fails, i.e. because it has fallen, the evaluation of the objective function is chosen inversely proportional to the elapsed time before the ground impact and weighted in order to be higher than the worst vehicle which be able to reach the finish line.

When the three-wheeler is able to follow the full trajectory, the objective function evaluates the vehicle dynamics quality. To be able to merge quality criteria of various nature into a single objective function $F_O$, we shape and size the criteria evaluations $E_i$ according to the following formula:

$$ F_O = \sum_{i=1}^{n_c} w_i E_i^{z_i} $$

where the weights $w$ and the exponents $z$ are determined to have an equivalence of satisfaction between all involved criteria. In fact, for all criteria, the user gives a satisfaction level which correspond to a "10 points" evaluation and an acceptance level which corresponds to a "1000 points" evaluation. For example, for the three-wheeler, the driver steering torque is part of the objective function: we consider that, physiologically speaking, the maximum acceptable value is 16 Nm but that 10 Nm is a satisfactory result.

4.3 Data conditioner

4.3.1 Algebraic constraints

The first constraints treated by the data conditioner are algebraic. Although all kinds of algebraic relations could be used to determine the so-called algebraic resulting parameter $p_{AR}$, we only use this first process to maintain symmetry and equality relations. The three-wheeler is, from a multibody point of view, symmetric with respect to the median plane. Thus, all data associated to left components are copied, with a possible change of sign, in the related right components data.

4.3.2 Geometrical constraints

Mechanical design and vehicle dynamics are studied simultaneously in the present work. This procedure generates numerous interdependencies between the two fields. Some mechanical design constraints appear as geometrical constraints that the optimization must consider. The equilibrium solution, that initially finds values of independent coordinates $q_{eq}$ (see Section 3.2), can be completed to solve —with the same solver— the geometrical constraints. There exist two complementary ways to do this:

- via a variable introduction: this method consists in adding the geometrical constraints equations $F_{ER}$ to the standard equilibrium Equations (8) and also an equivalent number of added equilibrium resulting parameter $p_{eER}$ to the solver variable list $x_{eq}$. For example, to impose the wheelbase $wb$ of the three-wheeler, we can set $F_{ER}(i) = wb_{current}(x_{eq}) - wb_{imposed}$ and incorporate the length of the rear fork into $x_{eq}$.

- via a variable exchange: this second method consists in replacing an exchanged independent generalized coordinate $q_{eu}$ in $x_{eq}$ by an exchanged equilibrium resulting parameter $p_{eER}$. To illustrate the interest of this second method, let us consider a standard equilibrium to determine the chassis height, given the neutral length of the front suspension spring. In
reality, from a mechanical design point of view, it is preferable to determine the neutral length as a result of the equilibrium, with the chassis height fixed at a desired value. The exchange between the neutral length and the generalized coordinate related to the chassis height in $x_{eq}$ permits to take this constraint into account simply.

In sum, the two methods can be used together. In such a case, the solver $S1$ solves the system:

$$F_r(x_{eq}) = 0$$
$$F_{ER}(x_{eq}) = 0$$

with

$$x_{eq} = q_u \setminus q_{eq} \cup p_{aER} \cup p_{vER}$$

Figure 7. The extended equilibrium scheme.

### 4.3.3 Modal constraints

As explained in the Section 3.3, we also consider modal characteristics as a measure of comfort. Imposing the frequency and the damping factor for both front and rear suspensions mode enables to optimize the vehicle without authorizing the optimizer to stiffen or soften the suspensions that can be favorable to decrease the objective function but has a major influence on comfort. By this way, we avoid to add comfort criteria to objective function and to implement a tire model coherent with uneven ground, which is CPU time costly but required to study the comfort. This simplification is fully efficient to save computational time with respect to a complete time simulation process.

In the case of the three-wheeler, this functionality has appeared to be essential because the stiffness of the front suspension has a direct influence on the energy needed to control the tilt motion, a criteria also included in the objective function.

As shown in Figure 8, a solver $S2$ drives modal resulting parameter $p_{MR}$ to meet modal criteria $F_{MR} = 0$ via the computation of an equilibrium, a modal analysis and via the selection of the suspensions modes at each iterative step.

Figure 8. The extended modal analysis scheme.
4.4 The reference simulation

For each particle tested, the algorithm runs a reference simulation based on a time integration as described in Section 3.4. During the simulation, the vehicle follows a reference track. This track must sufficiently excite the vehicle to evaluate all required performances. For the three-wheeler optimization, the chosen track is a "S" trajectory depicted in Figure 9. The first turn, to the right, has a maximum radius of 10 meters and is followed by a second turn, to the left, with a maximum radius of 8 meters. The initial speed of the three-wheeler is 7 m/s (~25 km/h).

4.5 Complete scheme

By combining the three previously described methods, it is possible to satisfy the three types of constraints. The global optimization scheme is depicted in Figure 10: it consists of four embedded algorithmic loops:

- the optimization loop, driven by the optimizer;
- the modal constraints loop, solved by $S2$;
- the geometrical constraints loop, solved by $S1$;
- the kinematic constraints loop, solved by $NR$ (not represented).
4.6 Parallel computation

Since the objective function evaluation via the trial is completely independent between particles, it is possible to run the optimization with a natural and easy parallelization of the tasks. In this respect, a CPU core is dedicated to the evaluation of each particle. These processes are called slaves. An additional core, the master, is reserved for the optimizer task. From a C-code implementation point of view, the MPI\(^4\) library is used to send \(x_o\) to a slave from the master and to get the objective function result.

The code execution is entrusted to the Lemaitre2 Cluster of the CECI\(^5\). The CPU type of the cluster is Westmere at 2.53 GHz. The cluster easily allows to allocate between 25 to 50 cores i.e. twice the number of treated \(p_o\).

4.7 Pros and cons

The main interest of the proposed approach is that all the constraints are not handled by the optimizer itself, but fully managed inside the trial function. At the same time, this allows us to decrease the number of optimized parameters. This way, each simulation in the Simulator block is run on a vehicle which satisfies all types of constraints. Thanks to the efficiency of the iterative solvers (S1 and S2), the computation time for the Data conditioner is negligible with respect to the simulation time (Simulator), leading to a really efficient optimization process.

A first drawback of the method is that the complete set of parameters must be correctly and manually partitioned into some subsets to ensure that the Data conditioner can solve the design constraints. Let us note that a numerical process could take over this task via a sensitivity analysis of the constraints equations with respect to the parameters.

As explained before, the Data conditioner is based on 3 embedded algorithmic loops. Despite the quality of the solvers, the convergence is not guaranteed for all set of optimized parameters, especially when the allowable ranges are large. The major issue is not to reach the solver convergence, because the currently treated vehicle is probably bad or impossible, but to give an evaluation to correctly guide the optimizer towards a better vehicle. This point must be improved in the next future.

5 RESULTS

The described methodology has been applied to the three-wheeler via five optimizations of respectively 3, 6, 9, 12 and 15 parameters. For all of them, we have requested 35 cores. Thus, the number of particles by generation was 34. Table 1 gives the main results and performances.

<table>
<thead>
<tr>
<th>Item</th>
<th>Number of parameters</th>
<th>Objective function</th>
<th>Number of generations</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>134009</td>
<td>110</td>
<td>00:36:26</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td>98962</td>
<td>176</td>
<td>01:20:56</td>
</tr>
<tr>
<td>3</td>
<td>9</td>
<td>90940</td>
<td>267</td>
<td>01:29:06</td>
</tr>
<tr>
<td>4</td>
<td>12</td>
<td>25072</td>
<td>469</td>
<td>01:47:51</td>
</tr>
<tr>
<td>5</td>
<td>15</td>
<td>25101</td>
<td>547</td>
<td>03:20:18</td>
</tr>
</tbody>
</table>

We observe the improvement of the evaluation as the number of parameters increases except for the last one. Of course, the decrease is more linked with the influence of the parameters with respect to the objective function than with their number. For example, from optimization 3 to 4, an important parameter which corresponds to the length of the top wishbone has been added: on can observe that the variation on the evaluation is significant.

\(^4\)Message Parsing Interface
\(^5\)Consortium des Equipements de Calcul Intensif (Belgium)
As expected, the number of generations and the elapsed time before convergence increases with the number of parameters. Due to the random nature of the optimizer, these performances are neither regular nor predictive. However, the global optimization globally appears as time efficient.

Figure 11. Optimization convergence.

Figure 11 shows the convergence for the five optimizations (left) and a zoom on the main interesting zone (right). On the top of the left figure, we can distinguish 3 plateaus which correspond respectively to candidates which fail in the Data conditioner, in the first turn and in the second turn (see Section 4.2). On the zoom, we can observe that the convergence is good although it is more and more difficult when adding parameters.

The evolution of the normalized parameter values for optimization 1 is shown in Figure 12. In this example, the 3 parameters are the orientation of the wheel axis with respect to the carrier, the length of the steering rod and the proportional coefficient of the titling controller. A similar evolution for the optimization 5 is given in Figure 13. The parameters change during a longer part of the optimization. Some parameters reach the top or bottom limits but most of them stabilize to an intermediate value.

Figure 12. Evolution of the 3 parameters during optimization 1.

As explain in Section 4.2, the objective function combines the evaluation of performances. Table 2 gives the optimal normalized value for two evaluators after each optimization. The first is the maximum steering torque during the simulation. The second is the mean of an evaluator which reflects a pseudo-Ackermann steering condition. Naturally, the evaluation of performances decreases with
the number of parameters. We remark that the addition of parameters with a strong influence be-
tween optimization 3 and 4 has allowed us to strongly decrease the second evaluator but to degrade
on the first and to push it to the acceptance limit.

Table 2. Evolution of two normalized evaluators

<table>
<thead>
<tr>
<th>Item</th>
<th>Steering torque</th>
<th>Pseudo-Ackermann</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>859,8</td>
<td>86426</td>
</tr>
<tr>
<td>2</td>
<td>547,8</td>
<td>47501</td>
</tr>
<tr>
<td>3</td>
<td>333,1</td>
<td>43980</td>
</tr>
<tr>
<td>4</td>
<td>966,1</td>
<td>511,9</td>
</tr>
<tr>
<td>5</td>
<td>1001,3</td>
<td>530,1</td>
</tr>
</tbody>
</table>

6 CONCLUSION

In this work, we have implemented a complete optimization method based on a multibody sim-
ulation and able to deal with a large number of parameters and design constraints. Thank to the
implementation which makes the most of each available numerical processes, the algorithm treats
algebraic, geometrical and modal design constraints with a high CPU time efficiency. Finally, the
method has been applied to a complex road vehicle —a tilting three-wheeler— and has allowed to
increase significantly its dynamic performances.

REFERENCES

domized Evolution Strategy with Covariance Matrix Adaptation (CMA-ES), Evolutionary
A Method to Combine an MBD Tire Model with a Thermo-dynamical one to improve the accuracy in the tire simulations

Francesco Calabrese*, Manfred Bäcker#, Axel Gallrein †,

* Fraunhofer Institut fuer Techno- und Wirtschaftsmathematik (ITWM)
Fraunhofer-Platz 1
D-67663 Kaiserslautern
francesco.calabrese@itwm.fraunhofer.de

# Fraunhofer Institut fuer Techno- und Wirtschaftsmathematik (ITWM)
Fraunhofer-Platz 1
D-67663 Kaiserslautern
manfred.baecker@itwm.fraunhofer.de

† Fraunhofer Institut fuer Techno- und Wirtschaftsmathematik (ITWM)
Fraunhofer-Platz 1
D-67663 Kaiserslautern
axel.gallrein@itwm.fraunhofer.de

ABSTRACT
Current state-of-the-art tire models may show a certain lack of accuracy in some advanced handling applications. This lack of accuracy is partly due to thermal effects. In reality, the tire rubber temperature can dramatically increase under certain conditions. The tire friction coefficient strongly depends on the temperature level. As a direct consequence of the temperature variations, the tire’s handling performance changes, e.g. when the temperature significantly differs from its optimal value, the tire’s grip level declines. As a result, the vehicle’s longitudinal and lateral behavior is influenced.

This paper shows that in order to increase the reliability of the tire models also in the described extreme conditions, it is necessary to couple a thermo-dynamical model with a mechanical one. The thermal model is important to estimate the temperature propagation inside the tire structure and the temperature evolution over time. It is shown how propagation and evolution is the result of a dynamic energy equilibrium between phenomena of different natures: heat is generated in areas with large cyclic deformations due to the energy dissipated from the rubber strains and in the sliding part of the contact patch due to sliding friction. The rubber cools down because energy is transferred to the air (internally and externally) and to the asphalt in the stick zone of the contact patch.

The described thermal model is designed to be used as a module and is applicable to tire models of various modeling details. In this paper, the coupling with an enhanced Magic Formula and with the detailed structural MBD tire model CDTire/3D are shown. The coupling strategy is more physical and direct in the case of the structural model, because of the local nature of this model. In fact, the energy input for the thermal model is calculated by using the tire structural model for each point of the tire structure volume. On the other hand, the structural model behavior is influenced by the locally calculated temperature by modifying the local structural properties such as shell dampings, stiffness’s and the local friction coefficient of the tread/asphalt contact. In the case of the Magic Formula, one challenge is to model the temperature creation and also the temperature transfer to the environment and also the temperature influence back to the Magic Formula itself, because of the missing physicality of the model.

At the end of the paper, the capabilities of the overall models are demonstrated and qualified in some illustrative tire and vehicle simulation scenario. The validation of the overall model will be shown using measured data from Formula 1.

Keywords: Thermodynamic, tire, flexible ring, vehicle dynamics, handling, NVH, comfort, safety
1 INTRODUCTION

The tire plays an essential role in car performance, handling, road holding and driving comfort. In the past many different tire models have been developed [1] to estimate forces and moments transmitted from the tire to the hub of the vehicle. For all these tire models the effect of the temperature on the tire behavior has not or only rarely been taken into account. But the effect of the temperature on the tire behavior is manifold.

Most adults may have experience driving a car with winter tires in late spring when the temperature is well over 20 °C. In such a situation one can really feel that the car tends to “swim” laterally on the highway or that the traction in curves become worse. The reason for this phenomenon can mainly be found in the softening of the tread block stiffness but also in the fact that the friction coefficient of the tread rubber leaves its optimum. So from this simple application case one can derive two main effects of the temperature on the tire:

• The structural stiffness’s (storage module) of the rubber compound is strongly temperature dependent, as well as the loss module.

• The friction between rubber and road is temperature dependent.

There is a third and also important temperature effect on the tire and that is the influence of the filler gas temperature to the tire inflation pressure. If one decreases the temperature of the inner air from 20 °C to 0 °C, the inflation pressure reduces approximately by 6 percent.

In the vehicle development processes, the temperature dependency of the tire behavior has not been considered very deeply in the past, while the effect on the inflation pressure may be covered by a temperature measurement of the filler gas, the local bulk or tread temperature cannot be measured with a reasonable effort. To get access to these temperatures a detailed thermo-dynamical tire model is needed to estimate the temperature creation and propagation in the tire.

Various authors [2] [3] [4] faced with this problem, they all agree in the physics description of the phenomena: the temperature evolution can be assumed as the result of the energy equilibrium between the air, the asphalt and the tire systems. The main terms of energy generation are friction and rubber dissipations, the main cooling terms are due to the external air and asphalt interaction. In [3] [4] the finite difference approach is fully described. In [2] [4] the authors propose an approach to couple the thermal model with semi-empirics tire models.

In this paper a full thermo-dynamical tire model will be introduced. The main purposes and unique contributions of the current study respect to previous ones are:

• The thermal model is designed like a module, in order to make possible to combine the thermal model with different mechanical tire models which can be vary in resolution, accuracy and physicality.

• The overall thermal tire model which will be introduced is a fully 3D finite volume based model with a fine resolution (which can be varied) in thickness, circumferential and lateral dimension. Respect to the finite difference one the finite volume approach is fully-stable and conservative.

• A new approach to interface the MF with the thermal module is analyzed. Respect to [2] it will be shown how the MF has to be expanded with a ‘contact patch model’. In this case also effect like lateral force, internal pressure, and contact patch shape will be considered in the energy distribution.

• The friction-grip dependency will be modelled in accord with the WLF law [5]

• The coupling with a fully 3d structural tire model [6] is described. In this case the structural energy dissipation and the material property are modelled in a detailed physical way respect to the first approach.
1.1 Importance of the thermal effect on tires and the mechanism of Temperature generation

Tires have a very complex composite structure made of different layers such as carcass, bandage and steel cords kept together with the filled rubber. The rubber compound has a prominent role for the tread behavior. Indeed due to its viscoelasticity properties the rubber is capable of dissipating energy in order to create friction and absorb impact.

Usually the heat generation due to the rubber strains is the most important source for temperature creation. At the same time as temperature arises, also the rubber loss and storage modulus decrease.

The rubber-asphalt interaction mechanisms are influenced as result of these phenomena because the rubber penetration into the asphalt asperities depends on the rubber compliance. Therefore the resulting friction coefficient is very sensitive to the temperature variation. This effect can be seen really clearly during a sweep angle experiment (See figure 1). In the following figures the temperature-force correlation during this kind of standard test is shown [2]. If the test is performed over a relatively long time (dark gray line – quasi-static) the temperature can also reach variations from 60 °C up to 80° C respect to the initial temperature (See figure 1). In this case it is possible to observe that the peak of the force characteristic curve changes when the temperature rises. Obviously these phenomena have an impact on the tire performance given that the friction is not always at its optimal level but changes with the temperature. If the test is performed in a relatively short time (gray line) the heat has not enough time to penetrate into the tread and warm up the tire, so in this case the temperature results are quite constant during the experiment, but the measurement is not suitable for parameter identification purposes because of the presence of the relaxation length effect in the slip stiffness.

Figure 1. Lateral force during a side slip angle variation for different time length of the experiment, right figure. Temperature during a side slip angle variation for different experiment durations, left figure.

Furthermore because of temperature creation and propagation there are also some secondary effects like the internal pressure and rolling resistance change. However in this article only the friction and inflation pressure dependency will be analyzed.

The phenomena responsible for the temperature variation of the rubber are basically the following [2] [3] [4]:
- Energy dissipated by the rubber strains
- Convection with the external air
- Friction in the sliding zone of the contact patch
- Conduction with the asphalt through the contact patch
- Propagation trough the tire volume
2 MODEL EQUATION DESCRIPTION

The heat conduction phenomena can be described by the Fourier diffusion equation (1). The equation can be integrated on the discretized volume of the tire (figure 2) using the so-called finite volume approach and applying the Navier-Stokes theorem (2)

\[
\frac{\partial}{\partial x} \left( k \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left( k \frac{\partial T}{\partial y} \right) + \frac{\partial}{\partial z} \left( k \frac{\partial T}{\partial z} \right) + \dot{q} = \rho c \frac{\partial T}{\partial t}
\]

(1)

\[
\int (\nabla \cdot k \nabla T + \dot{q}_g) dV = \int k \frac{\partial T}{\partial n} ds_i + \int \dot{q}_g dV = \int \rho c \frac{\partial T}{\partial t} dV
\]

(2)

Where the following terms can be called as:

- \( q_{tot} = \int k \frac{\partial T}{\partial n} ds_i \): Global thermal flux
- \( E_g = \int \dot{q}_g dV \): Energy generated inside the finite volume
- \( E_{st} = \int \rho c \frac{\partial T}{\partial t} dV \): Energy variation inside the material

The equation (2) obtained after the integration represents the energy equilibrium of a single finite volume element: In this equation the sum of the thermal fluxes \( q_{tot} \) that enter in the element through the surfaces with the internal energy generation \( E_g \) gives the variation of internal energy of the thermal cell \( E_{st} \) for an infinitesimal instant of time.

![Figure 2](image)

**Figure 2.** Representation of the thermal energy equilibrium of a single finite volume element.

The finite volume approach has the advantage over finite differences in that it allows using volume elements of arbitrary shape. This aspect is important because it allows having a non-uniform discretization without error in the energy balance calculation of the domain.

![Figure 3](image)

**Figure 3.** Example of discretization. NR, NTL, NCS are the three integer number that define how fine is the discretization along the thickness, the section and the circumferential direction.
The full volume of the tire is discretized along the circumferential, sectional and thickness direction. Such a fine discretization is necessary in order to also describe asymmetric scenarios of generation and propagation (e.g. large camber or full sliding condition during braking) and the various material properties along the thickness direction.

All the possible thermal phenomena in the tire thermodynamics can be included as flux generation at the boundary of the single thermal cell without changing the form of the equation (2). The basic form of each flux is represented as follow:

- Friction power density: it can be described as the product of the tangential force multiplied for the sliding velocity.

\[ \bar{q}_{f_p-i} = \tau_i \cdot v_s \]  

- Road Conduction: it depends on the difference in temperature between road and asphalt and on the thermal resistance. The thermal resistance depends on rubber-asphalt conductivities.

\[ \bar{q}_{rc} = R_{asphalt-rubber} (T_{road} - T_{rubber}) \]  

- Convection with the external gas: It depends on the difference of temperature between air and tire and on the convection coefficient. The convection coefficient \( H \) is calculated by using the Nusselt number (Nu). For this calculation the basic assumption is that the tire has the shape of a cylinder. The convection coefficient will be a function of the tire geometry, the relative velocity between tire and external air and of the air conductivity.

\[ \bar{q}_{ig} = H_f c (T_{ext-air} - T_{rubber}) \]  

- Heat generated by the rubber strains: It can be locally described as integral on the volume of the stress that multiplies the rubber strains. During the normal operational condition it will depend on the amplitude and frequency of the global forces \( F_x, F_y, F_z \) applied at the contact patch.

\[ \bar{q}_{g-i} = \frac{1}{V} \int_\Omega \sigma \cdot \epsilon dt dV \]

### 3 MODULARITY OF THE THERMAL MODEL

In this paragraph it will be described how the coupling between the thermal and the mechanical (tire) model can be achieved. Another important property of the model is the modularity. Readily available tire models as described in literature can be of various nature, semi-empirical (as the well-known Magic Formula), empirical or totally physical. The proposed thermal model can be applied like a module in combination with different kinds of tire model. In the scheme in figure 4 the general approach of the coupling concept is described:

**Figure 4.** Scheme of coupling between the thermal and the mechanical (tire) model
From the information coming from the tire model the thermal model creates several maps of
energy inputs for the thermal cells. The input energies are used to solve the energy equilibrium
described in the paragraph 2 and to calculate the temperature. Then the temperature is used to
modify some of the tire model properties like the internal pressure and friction coefficient (but
also stiffness and damping if the tire model is a fully physical structural model).

3.1 Friction-Temperature dependency

A very important part of the coupling concept regards the friction-temperature modification. As
described in the previous paragraph the friction varies with temperature. As outlined in various
article (Persson [5], figure 5) the William Landel Ferry law can be used to describe this kind of
variation. The adapted WLF principle says that the master curve of friction can shift in the
friction-frequency plane when the temperature changes. The shift value \( a \) can be calculated with
the WLF formula.

\[
\mu = A \exp \left( \frac{-E_a}{RT} \right) \exp \left( \frac{-C}{T} \right)
\]

\( \mu \) is the friction coefficient, \( A \) is a constant, \( E_a \) is the activation energy, \( R \) is the
ideal gas constant, \( T \) is the temperature, and \( C \) is a constant.

**Figure 5.** Example of friction coefficient as function of sliding velocity for three different temperature[6].

In the figure 6 is shown the effect of this coupling mechanism. A sweep slip angle experiment is
performed. In this case the thermal model is coupled with the Pacejka MF 2002. The lateral
force divided for the vertical load is plotted against the slip angle. In the figure 6 it is very clear
that the obtained effect of grip variation can be reproduced in quite an accurate way with this
approach.

**Figure 6.** Slip angle sweep test by using the thermo-mechanical model described in the article.
On the right side the temperature color-band. On the left side the FY/FZ versus slip angle plot.
Simulation time 40s. It’s clear how the temperature variation can have an influence on each
point of the characteristic shape due to the friction change.
3.2 Coupling with a full 3d structural model

The coupling with a fully 3d structural tire model is quite straight forward. In this case the CDTire/3D model is used [6].

Experimental and FEM studies [7] show that the tire energy dissipation are due to bending of the sidewall (20 – 30 %) and due to the belt and tread compression, bending and shearing (60 – 70%). Thanks to the 3D structural tire model [6] of all this behavior can be properly simulated. In this full 3D structural model the tire structure is described by using the shell Kirchhoff-Love formulation.

The entire anisotropic layers in the thickness are modeled. The coupling is straightforward because each energy loss due to strain of carcass, steel cords, deformation of the treads, sidewall and belt is available locally in the tire model and it can be passed to the thermal one for the energy balance calculation in the exact location in the thickness. The Temperature will be available for each location and can be used to modify the visco-elasto-plastic material properties of the structural tire model. In this way the whole model could also be used to make rolling resistance predictions.

In the figure 7 are shown the simulation results for a particular cornering condition with camber. It is possible to appreciate the deformation of the structure due to camber, lateral force (applied at the contact patch) and vertical load. The contact patch shape for example result conic for this particular extreme condition. Due to the particular deformed condition only the left side of the tire is affected from large temperature, this is because the large slide occurs in the left part of the belt (figure 7). It is clear from this example that also extremely complex non-symmetric thermo-mechanical condition can be described with this approach.

3.3 Coupling with a semi-empirical model (Magic formula)

The combination with a semi-empirical tire model like the Magic Formula it’s a little bit less exact than the previous one. The reason is that in this case no structural local information is
available and it is necessary to create them empirically. The two additional empirical functions used are a procedure for the contact patch shape evaluation and a model for the estimation of the global dissipated energy during tire deformation. Basically, the information about contact patch shape and position in the thermal volume domain are necessary to understand how to divide the finite volumes in elements in contact with the asphalt and elements cooled by the external air flux in order to distribute in a correct way the friction power, the heat absorbed by the asphalt and the heat taken by the air. Given that the tangential stress and the sliding velocity are not available locally in the MF, the original formulation (3) of the Friction Power has to be modified in the following way:

\[
\bar{q}_{fp-i} = τ_i \cdot v_x = R_{fp} \frac{(F_{x} v_{sx} + F_{y} v_{sy})}{A_{cp}}
\]

Where:
- \( F_x \) Global longitudinal force
- \( F_y \) Global lateral force
- \( R_{fp} \) Contact patch thermal resistance
- \( v_{sx}, v_{sy} \) Sliding velocities
- \( A_{cp} \) Contact patch area

Furthermore in this case is not possible to have the energy dissipated by the structure strains locally (as expressed in (6)) and another formulation is necessary to fill this other gap. As various authors suggest the whole energy dissipated by the structural deformation can be expressed as a function of the loads’ magnitude and frequency. The frequency can be assumed equal to the frequency of rotation. So the suggested local original formulation (6) becomes:

\[
\dot{q}_{g-i} = \int \sigma \frac{\partial E}{\partial t} dV = P(F_X, F_Y, F_Z, \omega)
\]

The formula (10) it’s an empirical formula and its parameters can be estimated with two strategies. One is by using rolling resistance measurements for various loads and rolling velocities at free rolling conditions. Another, by using a structural tire model (validated on a local basis) to get this information in a virtual way.

The MF is modified simultaneously using the scaling coefficient relative to the friction coefficient and the braking/cornering coefficients (\( L_{nu} \) and \( L_k \)). The scaling coefficients are used as function of the temperature, but the structure of the whole formula has not been changed.

After this adaptation, also the MF model can be used without losing so much accuracy in the thermal model simulation, as will be shown in the next paragraphs.
4 AN APPLICATION EXAMPLE

In the next paragraph the results of a simulation scenario will be shown and explained in detail. The time series kinematic inputs are represented in the following figure:

![Figure 9. Kinematic inputs of the simulation example.](image)

Some particular situations will be analyzed in order to validate the model: full sliding condition, pure rolling and slip angle sweep with camber different than zero.

![Figure 10. Energy and temperature results for the presented simulation scenario.](image)

In the figure 10 are plotted the input history of the single energies, the temperature generated on the surface (red line), in a medium layer of the treads (blue line, usually called bulk), and in the layer in contact with the internal air (green line usually called inner layer). In the first ten seconds the tire passes from pure rolling condition to the full braking. It is possible to observe how during this operation the energy generated from the friction power increases because of the longitudinal force. Meanwhile the energy generated by the rubber strains decay dramatically, because without rolling (the tire is skidding) the bending deformations of the belt are zero. On the other hand, because of the tire temperature increase,
also the differences between air temperature-rubber temperature and asphalt temperature-
rubber temperature became larger and as consequence, the same is also true for the relative 
energies.

Figure 11. Right: Tire model representation during the braking (instant 3s of the simulation). It’s possible to observe a gradient of temperature along the ring direction. This is because each section receive a different quantity of heat while the breaking is increasing. Instant 7s of the simulation. Left: Presence of an hot spot on the surf due to the skidding.

After the full braking in skidding condition it is possible to note a hot spot on the surface of the tire. The hotspot is created exactly in the zone that was sliding on the asphalt. The fact that this hotspot travels on the road periodically, with the same frequency of rotation, creates the oscillation of the asphalt absorbed energy that is visible around the second number ten in figure 11.

Later the tire is in pure rolling again and it’s cooled by the forced air flow (see figure 12) and asphalt contact. Then the temperature distribution becomes asymmetric because of the sweep angle excitation with a certain camber angle (see figure 12).

Figure 12. On the left the tire is cooled by asphalt and forced convection during the pure rolling. On the right side the temperature distribution is asymmetric during a slip angle excitation due to the camber.

The last observation regards the temperature evolution during the simulation. As it will be clear in the experimental data shown in the next paragraph the temperature seen on the tire surface is usually very different from the internal temperature of the tread. This happens because of the high thermal insulation properties of the rubber, which create a certain delay in the temperature propagation through the section thickness. It is indeed possible to observe this effect in the figure 13 (where the history of the propagation in a certain section is shown).

Figure 13. History of the propagation in a certain section at instant 2s (left picture) and 25s (right picture).
5 RESULTS: MBS SIMULATION AND FULL FORMULA 1 RACE DATA COMPARISON

In this paragraph the result of a full MBS vehicle are showed. The analysis regards a constant steering angle manoeuvre. In this manoeuvre the throttle position and the steering angle are constant. In the case of tires without thermal effect in the low slip area (linear tire force behaviour) the car will describe a perfect circle for each turn. If we add the thermal effect, due to temperature and cornering stiffness’s variation we will clearly see that for the same precedent condition the car will describe different elliptic trajectories for each turn (red line). If we try to increase more the velocity we will clearly see a drift effect due to the non-linear tire force characteristics. The fact that the trajectory will be different for each turn is a result of the temperature evolution and it would be never possible to reply this effect with a tire model not sensible to thermal effects.

Figure 14. MBS full vehicle simulation, constant steering angle maneuver for different velocities. The effect on the ‘thermal-tire’ are evident.

In the next figure the comparison between a lap of a Formula 1 tire (Sauber Motorsport) and the model is provided. The red line is the temperature measured on 5 points on the surface and the black is the temperature simulated by the model. The results find a good agreement with the data.

Figure 15. Comparison between F1 race measurement and the model.
6 CONCLUSIONS

A new thermo dynamical tire model has been introduced. The model has a finite volume based description. The resolution is scalable in all special dimensions. The temperature is predicted by solving the energy equilibrium at a local level. The thermal phenomena considered are the energy dissipated by the rubber strains, convection with the external air, friction in the sliding zone of the contact patch, conduction with the asphalt through the contact patch and the propagation through the tire volume. All these terms are considered to simulate the correct (dynamic) temperature creation and propagation in the tire.

The thermal model can be coupled with different kinds of mechanical tire models. In this article, the coupling with an advanced Pacejka MF, as well as with a physical 3D structural tire model (CDTire/3D) is analyzed. In the MF case, several enhancements have been introduced in order to make the coupling possible.

The thermal model resolution is independent from the used tire model. For both coupled tire models it is possible to reproduce the 3D distribution of temperature for various scenarios. The model results are very accurate especially in the reproduction of very non-uniform temperature distributions caused by large camber angles, large slip angles or full skidding situations. Moreover the estimated temperature from the thermal model has been used to introduce the friction dependency on temperature by using the WLF approach. The temperature creation and propagation, the friction-temperature dependency, and the thermal fluxes calculation (due to convection, conduction and friction) has been validated by using measurement done on F1 tires.

The validation results testify to the importance of the thermo-dynamical influence on the tire performance. The benefits in terms of phenomena reproduction and accuracy of the temperature and tire force estimations are proved.

The thermal model is real-time capable. This stays true also in the coupling with the temperature enhanced Pacejka MF. So beyond the usage of the model in offline CAE based vehicle optimization processes, the overall model can be used also as a SIL component in driving simulators.

REFERENCES

An Enhanced Tire Model for Dynamic Simulation Based on Geometrically Exact Shells

Michael Roller*, Peter Betsch†, Axel Gallrein*, Joachim Linn*

* Fraunhofer Institute for Industrial Mathematics
Fraunhofer Platz 1, 67663 Kaiserslautern, Germany

† Institute of Mechanics
Karlsruhe Institute of Technology (KIT)
Otto-Amman-Platz 9, 76131 Karlsruhe, Germany
Peter.Betsch@kit.edu

ABSTRACT

In the present work a tire model is derived based on geometrically exact shells. The discretization is done with the help of isoparametric quadrilateral finite elements. The interpolation is performed with bilinear Lagrangian polynomials for the midsurface as well as for the director field. As time stepping method for the resulting differential algebraic equation a backward differentiation formula is chosen. A multilayer material model for geometrically exact shells is introduced, to describe the anisotropic behavior of the tire material. To handle the interaction with a rigid road surface, a unilateral frictional contact formulation is introduced. Therein a special surface to surface contact element is developed, which rebuilds the shape of the tire.

1 INTRODUCTION

Acting as an interface between the car and the road, the tire model plays an important role in dynamic vehicle simulations. In commercial and scientific applications there exist several different modeling approaches for tires. When the tire model has to be embedded into a multi body system (MBS), lumped parameter models of varying complexity consisting of springs and dampers [10] are used, as well as simple data curve fits [13]. Very detailed but computationally demanding three dimensional finite element (FE) models are used for crash and misuse simulations [14]. A coupling of such 3D–FE tire models to MBS simulations is mostly not feasible due to the large number of degrees of freedom. Our purpose is to develop a continuum mechanical based structural model, which requires only modest amounts of computational resources so that a coupling with a MBS simulation is viable.

In [15] a tire model based on the geometrically exact shell theory of [16] was introduced, where finite elements (FE) are used for the spatial discretization [1]. The backward differentiation formula (BDF) was chosen as time stepping method for the arising differential algebraic equation (DAE) [3]. The developed tire model is able to handle pressure loads as well as frictionless contact with a rigid road surface. Also a special kind of orthotropic material is available, where one principal direction is parallel to the normal of the midsurface. The tire model is able to interact with a MBS simulation via co-simulation, with the rim forces and displacements as interface. In this work we enhance this discrete shell based tire model from [15] by adding a multi layered material model and frictional contact.

This work is structured as follows: First the equations of motion of the discrete shell are briefly derived in Section 2. After that, in Section 3 a multi layered material model is integrated in the shell model. In Section 4 the frictional contact formulation is presented as well as a special discretization of the contact surface. Typical tire test rig experiments are realized with the developed tire model in Section 5 and compared to reference solutions. At last we give a conclusion of our work in Section 6.
2 EQUATIONS OF MOTION

In this section we want to indicate how the equations of motion of the discrete shell are derived. This will be done in a concise way, for more details we refer to [15]. To describe the motion of the shell in space we follow the approach from [16]. Therein the configuration of the shell is given by two functions

\[ \varphi : \omega \to \mathbb{R}^3, \quad d : \omega \to S^2. \]  

Both are defined on a two dimensional parameter domain \( \omega \subset \mathbb{R}^2 \). The set \( S^2 := \{ d \in \mathbb{R}^3 \mid \| d \| = 1 \} \) represents the two dimensional unit sphere embedded in a three dimensional space. The function \( \varphi \) represents the midsurface of the shell and \( d \) a unit vector field pointing into the direction of the material fibers. Let \( h \) be the thickness of the shell, then its three dimensional configuration is given by

\[ \varphi (\mathbf{x}, \zeta) = \varphi (\mathbf{x}) + \zeta d (\mathbf{x}), \quad \varphi : \Omega \to \mathbb{R} \subset \mathbb{R}^3, \]  

with the domain \( \Omega := \omega \times [-h, h] \). The partial derivatives of the parameterization (2) define the covariant basis of the shell continuum \( g_{\alpha} := \varphi_\alpha \) and \( g_3 := \varphi_3 \). In this work greek indexes represent the numbers \( \{1, 2\} \), while small latin letters stand for the set \( \{1, 2, 3\} \). For partial derivatives the shortened notation \( \phi_\alpha := \partial \phi / \partial x_\alpha \) is used. The tangential vectors of the mid surface are defined as \( a_\alpha := \varphi_\alpha \). The contravariant bases of the shell is defined by the system of equations \( g_i \cdot \varphi^j = \delta^j_i \). The metrics of the mid surface is given by \( a_\alpha \cdot a_\beta = \delta^\alpha_\beta \). As additional invariants of the shell continuum a pseudo curvature \( \kappa_{\alpha\beta} := a_\alpha \cdot d_\beta + a_\beta \cdot d_\alpha \) and a shear field \( \gamma_\alpha := a_\alpha \cdot d \) are introduced. To measure the deformation of the shell continuum, a stress free reference configuration is defined by

\[ \phi_0 (\mathbf{x}, \zeta) := \varphi_0 (\mathbf{x}) + \zeta d_0 (\mathbf{x}), \quad \phi_0 : \Omega \to \mathbb{R}_0 \subset \mathbb{R}^3, \]  

where again \( \phi_0 \) represents the midsurface of the shell volume and \( d_0 \) is a unit vector field, see (1). To distinguish the reference configuration from the deformed configuration, we use capital letters for the quantities of the reference configuration. In the reference configuration it is additionally assumed that the director is perpendicular to the tangential vectors of the mid surface \( d_0 \cdot A_\alpha = 0 \). The deformation gradient of the mapping between the configurations \( \Phi := \phi \circ (\phi_0)^{-1} \) is given by \( F := \nabla \Phi = g_i \otimes \varphi^j \). As objective strain measure [2] we choose the Green-Lagrange tensor \( E = F^T F - Id \), which may be represented as \( E = E_{ij} (\varphi^j \otimes \varphi^k) \). The components of this strain measure \( E_{ij} := \frac{1}{2} (g_{ij} - G_{ij}) \) can be decomposed in differences of the single quantities of the two configurations

\[ E_{\alpha\beta} = \frac{1}{2} (a_{\alpha\beta} - A_{\alpha\beta}) + \frac{1}{2} (\kappa_{\alpha\beta} - \kappa_{\alpha\beta}), \quad E_{\alpha 3} = \frac{1}{2} (\gamma_\alpha - \Gamma_\alpha), \quad E_{33} = 0. \]  

Because of the unit length of the director there is no strain in the \( E_{33} \) component, which is called plain strain in literature [4]. As usual all terms witch are higher order one in \( \zeta \) are ignored.

The work conjugated stress measure [2] of the Green-Lagrange strain tensor is the second Piola-Kirchhoff tensor \( S = S^{ij} (G_i \otimes G_j) \). Both tensors are related through a material law. Because we only assume small local strains\(^1\), we restrict ourselves to a linear dependency between stress and strain. Therefore, it exists a fourth order tensor \( C := C^{ijkl} (G_i \otimes G_j \otimes G_k \otimes G_l) \), which describes this relation \( S^{ij} = C^{ijkl} E_{kl} \).

The components of the material tensor \( C^{ijkl} \) fulfill the usual symmetries [2]. To transfer this three dimensional material law to the shell, we additionally assume that the following components of

\(^1\)Because we choose the Green-Lagrange tensor as objective strain measure, rigid body motions of the whole shell do not contribute to the strain measure. Hence large rotations are still possible, which are necessary if we think about tire dynamics.
the material tensors vanish $C^{a\beta_{13}} = C^{a333} = 0$. So for the components of the strain tensor we get

$$S^{a\beta} = C^{a\beta_{13}} E_{1\pi} + C^{a333} E_{3\pi}, \quad S^{a\alpha} = 2C^{a3\beta_{3}} E_{\beta3}, \quad S^{33} = C^{33a\beta} E_{a\beta} + C^{3333} E_{333}.$$  \(5\)

By assuming plain stress ($S^{33} = 0$) we can rearrange (5) to

$$S^{a\beta} = H^{a\beta_{13}} E_{1\pi}, \quad S^{a\alpha} = \frac{1}{2} H^{a3\beta_{3}} E_{\beta3}.$$  \(6\)

Here $H^{ijkl}$ are the components of a shell material tensors $H = H^{ijkl}(G_i \otimes G_j \otimes G_k \otimes G_k)$, which are given by

$$H^{a\beta_{13}} = \frac{C^{a333} C^{33\pi}}{C^{3333}}, \quad H^{a3\beta_{3}} = 4C^{a3\beta_{3}}, \quad H^{3333} = 0.$$  \(7\)

### 2.1 Weak Formulation and Discretization

To solve the differential equation prescribing the motion of the shell continuum, we use the variational principle for the three dimensional problem [18] and eliminate the dependency of $\zeta$ by integration. The starting point of the problem is to find the function $\Phi$ such that for all variations $\delta\Phi$ the following holds

$$-\int_{\bar{\mathcal{R}}} FS : \delta F \, d\mathbf{X} + \int_{\mathcal{R}_0} F_0 \cdot \delta\Phi \, d\mathbf{X} = \int_{\mathcal{R}_0} \frac{d^2}{dt^2} \Phi \cdot \delta\Phi \, d\mathbf{X}.$$  \(8\)

The vector field $F_0$ represents the external load on the shell continuum, and $\rho_0$ the density of the reference configuration. In this special shell kinematics the variation of the deformation mapping is given by $\delta\Phi = \delta \phi \circ (\phi_0)^{-1}$ with corresponding functions

$$\delta\phi := \delta \phi + \zeta \delta d, \quad \delta \phi : \omega \rightarrow \mathbb{R}^3, \quad \delta d : \omega \rightarrow T_d S^2,$$  \(9\)

where $T_d S^2 := \{\delta d \in \mathbb{R}^3 | \delta d \cdot d = 0\}$ is the tangential plane of $S^2$ in $d$. In the following the function describing the shell configurations and its variation are summerized as $q = (\phi, d)^T, q_0 = (\phi_0, d_0)^T$ and $\delta q = (\delta \phi, \delta d)^T$.

With the transformation theorem and the surface measure $d\bar{\mathbf{X}} = \det(a_{\alpha\beta}) d\mathbf{X}$ the part of the internal work from (8) splits into two summands:

$$\int_{\Omega} S^{ij} \delta E_{ij} d\zeta d\bar{\mathbf{X}} = \int_{\Omega} E_{ab} H^{a\beta_{13}} \delta E_{1\pi} d\zeta d\bar{\mathbf{X}} + \int_{\Omega} E_{a3} H^{a3\beta_{3}} \delta E_{3\pi} d\zeta d\bar{\mathbf{X}}.$$  \(10\)

Therefore membrane stretching and bending have no influences on transverse shearing and vice versa. The first term of (10) is further split into

$$\int_{\Omega} E_{ab} H^{a\beta_{13}} \delta E_{1\pi} d\zeta d\bar{\mathbf{X}} = \int_{\Omega} \frac{1}{2} (a_{\alpha\beta} - A_{ab}) (H_{m}^{\alpha\beta_{13}} \delta a_{\pi}) + H_{c}^{\alpha\beta_{13}} \delta k_{\pi}) d\bar{\mathbf{X}}$$  + \int_{\Omega} \frac{1}{2} (\kappa_{ab} - \kappa_{a\beta}) (H_{h}^{\alpha\beta_{13}} \delta a_{\pi}) + H_{b}^{\alpha\beta_{13}} \delta k_{\pi}) d\bar{\mathbf{X}}.$$  \(11\)

With the approximation $G^{ij} = A^{ij}$ the variable $\zeta$ can be eliminated by integration, which yields

$$H_{m}^{\alpha\beta_{13}} := \int_{\frac{b}{2}}^{\frac{b}{2}} H_{m}^{\alpha\beta_{13}} d\zeta, \quad H_{c}^{\alpha\beta_{13}} := \int_{\frac{b}{2}}^{\frac{b}{2}} \zeta H_{c}^{\alpha\beta_{13}} d\zeta, \quad H_{b}^{\alpha\beta_{13}} := \int_{-\frac{b}{2}}^{\frac{b}{2}} \zeta^2 H_{b}^{\alpha\beta_{13}} d\zeta.$$  \(13\)

In the same way the integration of the second summand of (10) is executed. The two remaining parts of (8) are integrated over $\zeta$ in a similar way, see [15] for more details.
After the thickness variable is eliminated, we discretize the problem spatially, using the same quadrilateral FE approach as in [1]. Therein the mid-surface as well as the director field are interpolated locally by bilinear Lagrangian shape functions

\[ q^h = \sum_{f} q_f N_f, \quad q_f = (\varphi_f, d_f)^T \in \mathbb{R}^6. \] (14)

The discrete values \( q_f \) are positioned in the nodes of the FE-mesh. The same is done for the reference configuration \( q_0 \) and the variation \( \delta q \). Substituting these approximations into the integrated equation (8), we end up with a DAE for the motion of the discrete nodal variables \( q = (q_1, \ldots, q_N) \) of the shell model

\[ M\dot{q} = -R(q) + G^T(q)\lambda + F, \]  \quad (15)

\[ 0 = g(q). \]  \quad (16)

The constraint equation (16) and the corresponding force \( G(q)\lambda \) arise from the unit length condition of the director \( ||d|| = 1 \), which is enforced in the nodes of the FE-mesh. The equations (15)-(16) are integrated in time making use of the BDF-formula [3].

3 MATERIAL MODEL

In [15] a three dimensional orthotropic material model was transferred to the shell formulation. This is possible, if one principal direction of the material points into the normal direction of the mid-surface in the reference configuration. The variation of the material in the thickness direction due to the skew basis is neglected by the approximation \( G^{ij} \approx A^{ij} \). Therefore the components of the material tensor \( H^{ijkl} \) do not depend on \( \zeta \) any more and the integration is straight forward.

In this work we want to include a multi layered material construction in thickness direction. We assume that the layers are firmly connected. Let \( N \) be the number of layers in thickness direction. By \( h_l \) the height of the \( l \)-th layer is given. The thickness \( h \) of the whole shell continuum is given by the sum over all layers as: \( h = \sum_{l=1}^{N} h_l \). The material of each layer is assumed to be isotropic or orthotropic, like in [15]. Therefore each layer has its own material tensor \( C_{ijkl}^{ll} \) and could be transferred to a shell material \( H_{ijkl} \) as shown in (7). The coefficients of the material tensor of the whole shell continuum are reproduced by a piecewise defined function

\[ H_{ijkl}^{ll}(\zeta) = \begin{cases} 
H_{ijkl}^{ll}(\zeta + \frac{h}{2}) \in [0, h_1] \\
\vdots \\
H_{ijkl}^{ll}(\zeta + \frac{h}{2}) \in [h - h_{N-1}, h] 
\end{cases}. \] (17)

In each layer of the shell we make the approximation \( G_{ijkl}^{ll} \approx A^{ij} \), where \( G_{ijkl}^{ll} \) is the metric of \( l \)-th layer and \( A^{ij} \) those of the mid-surface of the whole continuum. Therefore we can examine the integration from (13) over the thickness on each layer and sum up the results

\[ H_{m}^{\alpha\beta\pi} : = \int_{-\frac{h}{2}}^{\frac{h}{2}} H_{ijkl}^{ll}(\zeta) d\zeta \sum_{l=1}^{N} h_l H_{ijkl}^{ll}, \] (18)

\[ H_{c}^{\alpha\beta\pi} : = \int_{-\frac{h}{2}}^{\frac{h}{2}} \zeta H_{ijkl}^{ll}(\zeta) d\zeta \sum_{l=1}^{N} \left( \frac{h_l^2}{2} + \left( -\frac{h}{2} + \sum_{j=1}^{l-1} h_j \right) h_l \right) H_{ijkl}^{ll}, \] (19)

\[ H_{b}^{\alpha\beta\pi} : = \int_{-\frac{h}{2}}^{\frac{h}{2}} \zeta^2 H_{ijkl}^{ll}(\zeta) d\zeta \sum_{l=1}^{N} \left( \frac{h_l^3}{3} + h_l^2 \left( \sum_{j=1}^{l-1} h_j - \frac{h}{2} \right) + h_l \left( \sum_{j=1}^{l-1} h_j - \frac{h}{2} \right) ^2 \right) H_{ijkl}^{ll}. \] (20)

These coefficients could be put into (12) without changing anything in the following discretization. If orthotropic materials should be used, two edges of each FE must by parallel to get a constant
orientation inside the element, see [15]. Then for each element and each of its layers the orientation of
the material is given by one constant angle. As the coefficients from (18)-(20) depend on the
reference configuration only, they can be computed in preprocessing.
In the tire each layer is made of reinforced material, which consists of two isotropic materials:
the matrix material \( (E_m, \nu_m) \) and the material of the fiber \( (E_f, \nu_f) \). The percentage amount
of reinforcements is given by \( V_f \), for the ratio of the matrix material directly follows \( V_m = 1 - V_f \).
With the help of the so called rules of mixture [9] the orthotropic material data of each layer could
be calculated. In this work we us the approach of Halpin and Tsai [8].

4 UNILATERAL FRICTIONAL CONTACT
We want to simulate the frictional contact interaction between a rigid road and the shell represent-
ing the tire. The contact surface of the road is usually given by a hight profile \( h : \mathbb{R}^2 \to \mathbb{R} \) as a two
dimensional surface

\[
X(s_1, s_2) = (s_1, s_2, h(s_1, s_2))^T .
\]

(21)

Its tangential vectors are given by \( T_\alpha := \frac{\partial X}{\partial s_\alpha} \), which define the normal vector of the surface \( N := \frac{T_1 \times T_2}{\|T_1 \times T_2\|} \). The contact surface \( \partial \mathcal{B}_c \) of the shell is given by a function

\[
x(\chi) = x(\varphi(\chi), \chi) , \quad x : \omega \to \mathbb{R}^3 , \quad \partial \mathcal{B}_c := x(\omega) ,
\]

(22)

which depends on the configuration of the deformed shell. We are using the concept of master-
and slave-surface, see [19]. Hence, for every point in the tire surface (slave) we are looking for
a contact point in the road surface (master). The closest distance point \( \tilde{X} \) of the road surface is
defined for a fixed \( x \in \mathcal{B}_c \) as

\[
||\tilde{X} - x|| \leq ||X(s_1, s_2) - x|| , \quad \forall s_1, s_2 .
\]

(23)

For each point of \( \mathcal{B}_c \) the distance function is defined as

\[
d_N(x) := (x - \tilde{X})^T \tilde{N},
\]

(24)

where \( \tilde{N} \) is the normal vector of the surface in the closest distance point. With this function we can
derive the KKT-conditions [19] for the frictionless contact

\[
p_N \geq 0 , \quad d_N \geq 0 , \quad p_N d_N = 0 ,
\]

(25)

where \( p_N \) is the normal contact pressure. To ensure (25), we introduce the normal gap function and its variation as

\[
g_N(x) = \begin{cases} 0 & d_N(x) \geq 0 \\ d_N(x) & d_N(x) < 0 \end{cases} , \quad \delta g_N = \begin{cases} 0 & d_N(x) \geq 0 \\ \delta x^T \tilde{N} & d_N(x) < 0 \end{cases} .
\]

(26)

The normal contact force is given by \( p_N = \varepsilon_N g_N(x) \) with a constant penalty parameter \( \varepsilon_N > 0 \). Hence, the contribution of the normal contact to the balance of momentum is given by

\[
\dot{\mathcal{B}}_N = \int_{\mathcal{B}_c} \varepsilon_N g_N \delta g_N dX = \int_{\omega} \varepsilon_N g_N \delta g_N \tilde{x} d\chi ,
\]

(27)

where \( \tilde{x} \) is the Jacobian determinant of (22).
4.1 TANGENTIAL CONTACT

To compute the frictional contact between the tire and the road, we have to evaluate the path $g_T$ of a particle in contact on the *master*-surface. Hereby two cases are distinguished: Either the particle is sticking on the surface, hence, there is no relative movement between both surfaces in this point $\dot{g}_T = 0$, or the particle is sliding on the surface, hence, a reaction force proportional to the normal contact force is acting on the particle $t_T = \mu p_N \frac{\dot{x}_T}{\|\dot{x}_T\|}$, where $\mu$ is the coefficient of friction.

To handle both cases, a method from theory of plasticity is used [18]. The glide path is divided in a reversible elastic $g^e_T$ and a irreversible plastic part $g^p_T$, such that $g_T = g^p_T + g^e_T$ holds. The elastic part represents sticking. With a penalty stiffness $\varepsilon_T > 0$ the corresponding traction force is given by

$$t_T = \varepsilon_T g^e_T .$$

The plastic part of the glide path is given due to an evolution equation. This equation is derived by maximizing the dissipative power of sliding with respect to the traction force $t_T$, which is additionally restricted by the condition

$$f_C(t_T) := \|t_T\| - \mu p_N \leq 0 .$$

The solution of this optimization task is given by the following problem: Find $t_T$ and $\lambda_f$ such that the equations

$$g^p_T = -\lambda_f \frac{\partial f_C}{\partial t_T} ,$$

$$\lambda_f \leq 0 , \quad 0 \geq f_C(t_T) , \quad \lambda_f f_C(t_T) = 0 ,$$

hold, where (31) are the KKT-conditions of the tangential contact. The discrete evolution of the plastic part of the glide path and the discrete traction forces are evaluated by means of the radial return algorithm [7].

The variation of the slip path $g_T$ can be obtained by differentiating the optimality condition of the nearest contact point (23) with respect to time, see [17]. Since the road surface is not in motion and we assume that the normal penetration into the road surface is negligibly small, this variation is given by

$$\delta g_T = (I - \tilde{N} \otimes \tilde{N}) \delta x .$$

Hence, for the contribution of the tangential contact to the balance of momentum we get

$$\tau_T = \int_{\beta_T} t_T \delta g^T_T \delta x = \int_{\omega} t_T \delta g^T_T \tilde{x} d\chi .$$

4.2 DISCRETIZATION OF THE CONTACT SURFACE

To incorporate the contact force in the discrete equation of motion (15)-(16), the integrals (27) and (33) must be discretized. This could be done with a node to segment approach for example, where the contact is evaluated in the nodes of the FE mesh. As discussed in [15], this approach is not able to detect small obstacles. Therefore, the discretization of the FE formulation with bilinear interpolation inside the element was used, and the contact was evaluated at the Gaussian integration points. In this approach the circumferential discretization degenerates to a polygon with a discrete number of edges. This results in a drawback if the tire is rolling on flat ground: In Figure 1 the problem is visualized by the comparison of a rolling rigid polygon and a rigid circle. To eliminate this problem, we use the discrete director in addition to the midsurface points to create the element wise contact shape. This should be done such that the tangential vectors of the
surface are perpendicular to the discrete directors in the nodes of the FE mesh. In [11] such an interpolation is realized with a biquadratic shape function. This approach has the drawback that no turning points inside the element could be realized. This may result in strange contact shapes for some situations [12]. To circumvent that problem, we choose the bicubic Hermite-Polynomials as local shape functions for the contact surface. We will explain the procedure on a one dimensional example, where two points $x_\alpha$ with normal information $d_\alpha$ should be interpolated.

The cubic Hermite polynomials are given over the interval $[0, 1]$ by

$$
H_1(s) = (1 + 2s)(1 - s)^2, \quad H_2(s) = s^2(3 - 2s), \quad H_3(s) = s(1 - s)^2, \quad H_4(s) = s^2(s - 1).
$$

Therefore the interpolation of the curve reads

$$
x(s) = x_1 H_1(s) + x_2 H_2(s) + m_1 H_3(s) + m_2 H_4(s). \quad (35)
$$

At the end nodes of the interpolation curve we obtain by straightforward computation

$$
x(0) = x_1, \quad x(1) = x_2, \quad x'(0) = m_1, \quad x'(1) = m_2. \quad (36)
$$

Therefore $m_\alpha$ must be chosen such that $m_\alpha \cdot d_\alpha = 0$ holds. With the help of the rotation matrix

$$
R(d_\alpha, b) := \frac{||b||}{||d_\alpha \times b||} (\text{Id} - d_\alpha \otimes d_\alpha) \in \mathbb{R}^{3 \times 3} \quad \text{with} \quad b = x_2 - x_1, \quad (37)
$$

the tangential vectors are given by

$$
m_\alpha = R(d_\alpha, b)b, \quad (38)
$$

and $m_\alpha \cdot d_\alpha = 0$ holds obviously. Additionally the length of the tangential vectors are equal to the distance between the interpolated points $||m_\alpha|| = ||b||$. If this approach is used locally to construct a curve from many given points and normals, the direction of tangential vectors of resulting local curves are equal in the nodes. Nevertheless, its magnitudes could differ. Therefore, the curve is not continuously differentiable.

This approach can be easily transferred to a two dimensional surface according to the concept of a tensor product surface, where each edge is interpolated according to the one dimensional approach. To do that a bicubic shape function is used

$$
x(s, t) = \sum_{i=1}^{4} \sum_{j=1}^{4} C_{ij} H_i(s) H_j(t). \quad (39)
$$

Additionally the mixed derivatives in the nodes of the surface could be defined. We set them to zero $C_{34} = C_{43} = C_{33} = C_{44} = 0$, which correspondence to the so called Ferguson patch [5]. In Figure 2 the interpolation of such a surface is visualized with the corresponding coefficients from (39).

With this approach the integrals (27) and (33) are element-wise discretized. For the interpolation of the variation $\delta x$ no tangential or normal information is used. The contact algorithm is evaluated at the discrete Gaussian integration points as in [15].
5 TIRE SIMULATION

Finally we want to show some classical tire test rig simulations with our model constructed from geometrically exact shells. As comparison we choose the discrete finite-difference based multi layer shell model CDTire/3D [6]. This model has been parametrized such that the experimental results of a real tire on a test trig could be reproduced. Therefore, we may use CDTire/3D to perform numerical experiments for benchmarking. As inner pressure of the tire in all simulation 2.7[bar] is chosen. As friction coefficient $\mu = 1$ is assumed.

First we want to calibrate the material data of the tire model presented in this work. To do that we choose a simple vertical stiffness simulation of a tire. Therein the tire is flattened out against a flat rigid surface by deflecting the rim in vertical direction. This implies a vertical reaction force on the rim. The relation between deflection and force produces a characteristic curve of the tire.

The net of discrete nodes of the midsurface are given by cylindrical revolution of the tire cross section. The directors are chosen heuristically such that they approximate the normals of the geometrically simple reference surface. The corresponding number of elements in the cross section are 16 and 50 in circumferential direction. A tire is build from different layers of cord reinforced sheets, each of which can be considered a functional layer. Following the natural concept of parameterizing these individual functional layers [6], we are now using the multi layered material model described in Section 3. The tread is segmented in six layers. The matrix material of every reinforced layer is rubber. Starting from inside, the tire the first layer purely consists of rubber, which is modeled as an isotropic material. The second layer is called carcass, its reinforcements are made of synthetic fibers. They are oriented perpendicular to the direction of rotation. In the third and fourth layer steel-cords are used to strengthen the material. The steel wires are oriented mirror-symmetric in both layers concerning the direction of rotation. The fifth layer is called cap ply. It is reinforced with nylon cords, which are directed along the direction of rotation. A layer consisting purely of rubber is the last of the six layers. For the first and the last layer, we choose a isotropic material model. For the others an orthotropic material model is chosen, whose material coefficients are computed by the approach of Halpin and Tsai [8], using the material data from Table 1 with the angle $\alpha$ measured with respect to the direction of rotation. For the sidewall the two steel-cord layers and the cap-ply are substituted by layers with the same thickness consisting of isotropic rubber.

The material informations of Table 1 are adopted from a datasheet of a tire. The elastic modulus of the rubber matrix and the synthetic fibers in the carcass are optimized such that the result of vertical stiffness simulation coincides with the reference solution, see Figure 3. The optimization was performed fully automatized, by minimizing the squared distance between the two curves.
Table 1. Optimized material properties of the different layers of the tire.

<table>
<thead>
<tr>
<th>Material</th>
<th>$E_m\ [N/m^2]$</th>
<th>$\nu_m$</th>
<th>$E_f\ [N/m^2]$</th>
<th>$\nu_f$</th>
<th>$V_f$</th>
<th>$\alpha\ [\deg]$</th>
<th>$h_i\ [m]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rubber</td>
<td>$7.39 \cdot 10^6$</td>
<td>0.5</td>
<td>$3.9 \cdot 10^5$</td>
<td>0</td>
<td>0.26</td>
<td>90</td>
<td>$1.55 \cdot 10^{-5}$</td>
</tr>
<tr>
<td>Carcass</td>
<td>$7.39 \cdot 10^6$</td>
<td>0.5</td>
<td>$1.33 \cdot 10^8$</td>
<td>0</td>
<td>0.17</td>
<td>-24</td>
<td>$1.04 \cdot 10^{-3}$</td>
</tr>
<tr>
<td>Steel-cord</td>
<td>$7.39 \cdot 10^6$</td>
<td>0.5</td>
<td>$1.98 \cdot 10^{11}$</td>
<td>0</td>
<td>0.17</td>
<td>24</td>
<td>$0.9 \cdot 10^{-3}$</td>
</tr>
<tr>
<td>Steel-cord</td>
<td>$7.39 \cdot 10^6$</td>
<td>0.5</td>
<td>$3.43 \cdot 10^9$</td>
<td>0</td>
<td>0.26</td>
<td>0</td>
<td>$0.65 \cdot 10^{-3}$</td>
</tr>
<tr>
<td>Cap-ply</td>
<td>$7.39 \cdot 10^6$</td>
<td>0.5</td>
<td>$3.9 \cdot 10^5$</td>
<td>0</td>
<td>0</td>
<td>2.5</td>
<td>$2.5 \cdot 10^{-3}$</td>
</tr>
</tbody>
</table>

![Figure 3](image_url). Reaction force on the rim, due to the flattening of the tire against a flat rigid road.

5.1 LATERAL AND LONGITUDINAL STIFFNESS

We want to have a look at the lateral and the longitudinal stiffness of the tire in the following simulation. Therefore, the tire is deflected against a flat rigid surface until a vertical reaction force of $4\ [kN]$ is reached. In the lateral stiffness simulation the rim is deflected perpendicular to the running direction of the tire. This implies a lateral reaction force on the rim. In the longitudinal stiffness simulation the rim is deflected in the running direction of the tire, while all rotations are locked. Therefore a longitudinal reaction force is acting on the rim. Again the results of the simulation are compared with those of CDTIre/3D.

In Figure 4 the simulation results of the lateral stiffness simulation are visualized. The lateral deflection of the rim is plotted against the reacting force. After the lateral force reaches the size of the vertical force the tire is sliding with a constant reaction force in lateral direction over the flat ground. Because the simulation results of the tire model developed in this work match with those of CDTIre/3D, the lateral stiffness of the tire is predicted correctly with our model without doing any additional parameter modification.

The results of the longitudinal stiffness simulation can be seen in Figure 4. Therein the longitudinal deflection of the rim is plotted against the reaction force. The tire again starts to slide on the surface with a constant reaction force, if the longitudinal reaches the vertical force. At the beginning of the simulation the whole contact patch is sticking on to the road surface. In this situation both curves match pretty well. This indicates that the structural stiffness of the tire is predicted in a correct way. When the contact points start to slide, both curves do not match perfectly together. This could be because the forces in the contact patch are not predicted accurate enough. However, this results are again achieved without any further adjustments on the parameters.
5.2 LATERAL SLIP

In this section we want to show a first dynamical simulation. The rim is moved with a constant velocity of 10\(\text{m/s}\) in running direction of the tire over a flat surface with a constant vertical deflection, such that a reaction force of 4\([kN]\) is acting on the rim. The rotation of the rim due to its longitudinal movement and contact is realized via a co-simulation of the tire model and the rim, see [15]. The duration of this simulation is five seconds. In this time the rim is rotated about its vertical axis to simulate cornering.\(^2\) The time signal of the corresponding angle is given by

\[
\alpha(t) = \begin{cases} 
\frac{15\pi}{180} \sin \left( \frac{2(t-0.5)}{\pi} \right) & 0.5 \leq t \leq 4.5 \\
0 & \text{otherwise}
\end{cases}
\] (40)

In the literature this angle is called slip angle. Due to cornering the tire produces a lateral force on the rim. In Figure 5 the simulation results are visualized. Therein the lateral force is plotted against the slip angle \(\alpha\), which results in a curve that is characteristic for a tire. The simulation result of the tire model developed in this work matches favorably to the reference solution. So, we

\(^2\)The vehicle drives through a curve.
predicted the cornering behavior of the tire quite well without doing any adjustment of the model parameters. In Figure 5 a hysteresis could be seen because of the movement of material point through the contact patch.

5.3 LONGITUDINAL SLIP

Finally we want to simulate the accelerating and breaking of a tire. To do that, a longitudinal slip experiment is realized. Therein, the rim is moving with a constant velocity in rotational direction, while the rotation velocity of the rim is varied in time. To examine the simulation the rotation velocity $\omega$ of free rolling $^3$ tire must be evaluated. The longitudinal slip $\kappa$ is defined as relative difference of the current rotational velocity $\omega$ and free rolling velocity $\omega_0$

$$\kappa := \frac{\omega - \omega_0}{\omega_0}. \quad (41)$$

The simulation is executed in the time interval $[0,5]$. In contrast to the lateral slip simulation in this experiment also the rotation of the rim is prescribed. This is done via a time signal for the longitudinal slip, which implies the rotational velocity

$$\omega(t) = (1 + \kappa(t)) \omega_0 \quad \text{with} \quad \kappa(t) = \begin{cases} \sin \left(\frac{2(t-0.5)}{\pi}\right) & 0.5 \leq t \leq 4.5 \\ 0 & \text{else} \end{cases}. \quad (42)$$

Because of the different model approaches, both simulated tire models have different rotation velocities $\omega_0$ for the same translation velocity $v = 10[m/s]$. However, because the longitudinal slip is defined relative to $\omega_0$, we can compare both models.

In Figure 6 the results of both simulations are visualized. Therein the longitudinal slip is plotted against the longitudinal force. Both simulation results match quite well. Similar as in the longitudinal stiffness simulation differences occur, when the particles in the contact patch start sliding. Due to the stick and slip history of the material points moving through the contact patch, there is a hysteresis in the graph in Figure 6 similar as in the lateral slip simulation.

6 CONCLUSION AND OUTLOOK

In this work the tire model from [15] is improved by a multilayer material model and a more complex contact model, which includes lateral contact. With this enhanced model it is possible to predict the dynamical behavior of a tire, while the adjustment of the parameters is done only via a simple vertical stiffness experiment.

$^3$Free rolling is defined such that no torque is acting on the rim about the rotation axis of the tire.
REFERENCES


Estimation of influence of stiffness of chassis frame on LCV dynamics in conditions of curvilinear motion

Anton V. Tumasov, Sergey Y. Kostin, Danila A. Butin, Aleksey A. Vasiliev, Andrey A. Vashurin, Yuri P. Trusov

Transport Systems Institute
Nizhny Novgorod State Technical University n.a. R.E. Alekseev
Minin str., 24, 603950 Nizhny Novgorod, The Russian Federation
anton.tumasov@nntu.nnov.ru

ABSTRACT

The paper deals with the problem of calculative and experimental studying of influence of torsional stiffness of chassis frame on LCV dynamics in conditions of curvilinear motion: start of the corner and line changing. The object of research is LCVs with cargo bed (the vehicles capacity is up to 3,5) that have a wide range of wheel base dimensions and, as a result, different torsional stiffness of chassis frame. The influence of torsional stiffness of chassis frame on LCV dynamics is studied on the basis of results of experiments and simulation. The static experimental research was made in laboratory environment where the stiffness of the chassis frame was estimated. Road tests were made in accordance with the Russian Standard GOST R 52302-2004 that presupposes different types of tests: static rollover, dynamic maneuvering on a road (line changing and running into the corner). The multibody simulation was made in MSC.ADAMS/CAR and MSC.NASTRAN software that allows imitating the LCV dynamics considering flexibility of chassis frame.

Keywords: LCV, vehicle dynamics, frame torsional stiffness, curvilinear motion, chassis flexibility.

1 INTRODUCTION

The initial stages of this study were presented at two international conferences FISITA 2014 (F2014-IVC-048) and ASME 2014 (AVT, DETC2014-34641), [1]. These studies presented the combination of experimental and simulation methods that could be used for estimation of LCV active safety characteristics (first of all cornering stability). The experimental method of estimation of cornering stability was based on regulations of the Russian Standard GOST R 52302-2004 that presuppose static and dynamic vehicle testing. The multibody simulation method was based on MSC.ADAMS/CAR software capabilities. The approval of developed LCV multi-body model was made on a basis of good correlation between simulation results and experimental data. The relationship between LCVs design parameters (axle load distribution, height of the center of gravity, vertical and angular suspension stiffness) and active safety characteristics were received.

Paper [1] presented results of simulation that was made with using of multibody LCV model where a frame, a cabin and a cargo platform were presented by absolutely rigid bodies. Only one modification of LCV was studied (basic modification with wheel base 3145mm).

It is worth to underline that a huge range of modifications could be developed on the basis of one LCV chassis (the basic chassis). Each modification, in turn, can be reconstructed in a way that allows LCV to carry out all necessary functions most effectively. As an example on Figure 1 long wheel base LCV modifications are shown. Such kind of vehicles have the same construction as basic LCV has, but the wheel base is longer and allows to carry long objects. The gross weight of such modifications is similar to basic LCV (the carrying capacity is lower, because of additional mass of frame extender), but longer wheel base became a reason of lower
torsional stiffness of the frame. As a result a performance of such vehicles in wide range of maneuver could be really different.

![Figure 1. Long wheel base LCV modifications](image)

Torsional stiffness of chassis frame could have a strong effect on LCV dynamics in conditions of curvilinear motion (line changing or running into the corner). The multibody simulation could be used for estimation of influence of frame stiffness on LCV dynamics.

The assessment of the impact of various vehicles design parameters on its steerability and stability properties was explored in papers of different authors [2 – 6], but in most cases the research made with models that consist of rigid frame (without possibilities to be deformed under action of dynamics loads).

This study is a continuation of research that is presented in paper [1] and pays a key attention to:

- Estimation of influence of wheel base on torsional stiffness of chassis frame;
- Analysis of mode of deformation of chassis frame in dependence of wheel base length;
- Estimation of influence of torsional stiffness of chassis frame on LCV dynamics.

## 2 LCV FRAME TORSIONAL STIFFNESS

At the first stage of the study the verification of LCV FEM frame was made. The simulation was done in MSC NASTRAN software (Figure 2a). The experiment research (Figure 2b) was made on a special testbench that allows imitating of twisting conditions.

The comparison analysis of results (Table 1) shows relatively small divergence between strains that were measured in control points (Figure 2c) during simulation and real static test. It gives the ground to conclude that the FEM of LCV frame has an adequate behavior in twisting conditions. This FEM as well as finite elements model of a cab and cargo platform were used in multibody simulation that was carried out with the help of MSC.ADAMS/CAR software.
Four modifications were studied: one basic variant (wheel base 3145 mm) and three variants with longer wheel base: 3754 mm (+600 mm); 4445 mm (+1300 mm); 5045 mm (+1900 mm).

The research of influence of wheel base length on torsional stiffness of the structure (frame with cab and cargo platform – Figure 3a) was made. The diagram of Figure 3b shows the stiffness decay in dependence on increasing wheel base.

In parallel the analysis of mode of deformation of chassis frame in dependence of wheel base length was performed. Table 2 shows the results of simulation. It could be seen that longer wheel base lead to the lower value of natural-vibration frequency.

### 3 MULTIBODY SIMULATION MODELS

Four multibody models were developed (Figure 4), each of them include: front independent and rear dependent suspensions; steering components; flexible frame, cabin and cargo platform; the rigid body containerized cargo with mass-inertia parameters. It is worth to underline that the
position of cargo was chosen in a way that allowed having the same mass distribution among front and rear axis for each model. The gross weight (3500 kg) was also similar for all models.

**Figure 3.** Calculative estimation of structure (frame, cab and cargo platform) torsional stiffness

**Table 2.** Natural-vibration frequency, Hz

<table>
<thead>
<tr>
<th>№ of mode</th>
<th>Shape</th>
<th>Wheel base, mm</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>3145 + 600</td>
</tr>
<tr>
<td>1</td>
<td><img src="image1" alt="Image" /></td>
<td>6,07</td>
</tr>
<tr>
<td>2</td>
<td><img src="image2" alt="Image" /></td>
<td>20,98</td>
</tr>
<tr>
<td>3</td>
<td><img src="image3" alt="Image" /></td>
<td>25,22</td>
</tr>
<tr>
<td>4</td>
<td><img src="image4" alt="Image" /></td>
<td>32,63</td>
</tr>
<tr>
<td>5</td>
<td><img src="image5" alt="Image" /></td>
<td>33,13</td>
</tr>
</tbody>
</table>
All models were virtually tested in several conditions:

- Lateral static stability test (Figure 5) where model mounted on the rollover platform. The angle of a rotary platform measured at a moment when cornering breakaway of both outer wheels observed.
- “Start of the corner” test (Figure 6) that define the indicators characterizing stability of a vehicle in critical modes of movement on a curvilinear trajectory. The maximum speed of a vehicle when performing maneuver is determined in case of crossing the corridor marking or cornering breakaway.
- "Line changing" test (Figure 7) is intended to determine the maximum speed of maneuver at changing the line on a limited track section. Conditions of this virtual test are similar to conditions of the "start of the corner" test.
4 SIMULATION RESULTS

The results of multibody simulation are shown in the Table 3. It shows that the longer the wheel base (the less the torsional stiffness) the lower the values of LCV stability parameters (the rollover angle and the maximal speed of curvilinear maneuver).

Table 3. The results of multibody simulation

<table>
<thead>
<tr>
<th>Virtual test</th>
<th>Parameter</th>
<th>Model variant (wheel base, mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>3145</td>
</tr>
<tr>
<td>Lateral static stability test</td>
<td>Rollover angle, grad</td>
<td>38,2</td>
</tr>
<tr>
<td>“Start of the corner” test</td>
<td>Max. speed of maneuver, km/h</td>
<td>63,2</td>
</tr>
<tr>
<td>&quot;Line changing&quot; test</td>
<td>Max. speed of maneuver, km/h</td>
<td>74,6</td>
</tr>
</tbody>
</table>

5 CONCLUSIONS

– The increasing of LCV wheel base length (up to 30...60%) reduce the torsional stiffness of LCV structure approximately up to 30...40%.

– Torsional stiffness of chassis frame has an efficient influence on LCV dynamics in conditions of curvilinear motion: decreasing of LCV torsional stiffness up to 30...40% reduce maximal speed of curvilinear maneuver up to 10...12% for “Start of the corner” test and up to 12...17% for “Line changing” test.

– Less torsional stiffness of LCV structure lead to the less static rollover angle (less vehicle static stability).

6 ACKNOWLEDGEMENTS

The research was done with the financial support of Ministry of Education and Science of the Russian Federation in the frame of the complex project “The establishment of the advanced technology production of ecological and resources-economy LCV” under the contract №02.G25.31.006 from 12.02.2013 (Governmental Regulation №218 from 09.04.2010).

The experimental research was conducted with the use of measurement equipment of the NNSTU Center of collective using “Transport Systems”.

Figure 7. "Line changing" virtual test
REFERENCES


Estimation of manoeuvrability of multi-axis tractor-lorry-trailer combination for transportation of large-dimensioned indivisible loads

Anton V. Tumasov*, Lev V. Barakhtanov*, Sergey E. Manyanin*, Aleksey A. Vasiliev*
Sergey L. Tropin#

* Transport Systems Institute
Nizhny Novgorod State Technical University n.a. R.E. Alekseev
Minin str., 24, 603950 Nizhny Novgorod,
The Russian Federation
anton.tumasov@nntu.nnov.ru

# Joint Stock Company “Spectyazhavtotrans”
Yablochkova str., 29b, 127322, Moscow
The Russian Federation
info@statnn.ru

ABSTRACT
The paper describes the peculiarity of calculative and experimental estimation of manoeuvrability of multi-axis tractor-lorry-trailer combination intended for transportation of large-dimensioned indivisible loads. Such kind of specialized vehicles are used for freightage of indivisible units, machinery and huge equipment for industrial facilities; complete indivisible building blocks and structures and etc. The safety of such complicated transportation process depends on vehicles construction as well as rational planning of goal course considering the tractor and trailer manoeuvrability. The paper presents the mathematical equations that describe the behavior of multi-axis tractor-lorry-trailer combination in condition of curvilinear motion that could be used for simulation of possible tractor and trailer trajectories in wide range of manoeuvres. The results of simulation are described as well as procedure of real tests on a road. It was found that simulation results have good correlation with experimental data. The key attention of the paper is pay to analysis of tractor and trailer trajectories and estimation of possibility of safety movement in real road infrastructure.

Keywords: multi-axis tractor-lorry-trailer combination, heavy-duty trucks, large-dimensioned indivisible loads, manoeuvrability.

1 INTRODUCTION
While transportation of the most common goods by road is performed by simple trucks or truck-trailer combinations, the most cases of transportation of oversized and heavy cargoes (OHC) needs exclusively special heavy road cars which include heavy road trucks or off-road tractors. Features of heavy-duty trucks define their fundamental difference from the general-purpose vehicles. In relation to such vehicles parameters of a tow unit have a dominant importance as they determine: load capacity, axial load, cargo dimensions, maneuverability and other main vehicles performance properties.

The process of choosing of a vehicle type for OHC implies that for a heavy trailer is needed to be selected an appropriate truck that would provide possibility of transportation in predetermined operating conditions. It is needed to be mentioned that such important for ordinary trucks characteristics as speed and dynamics are of secondary importance for OHC vehicles. At the forefront indicators that determine traffic safety.

Taking into account large sizes and high axial loads, usage of multi-axle vehicles of large and extra-large load capacities raises the issue of maneuverability on road turns, off-road, building sites and at other road conditions.

The importance of maneuverability is also confirmed by the fact that the goods are often have to be delivered at destination points without additional capital investments in road improvement.
Obviously, in order to increase road safety during the transportation of OHC it is necessary to develop relevant instruments for solving of such problems at the stages of vehicle design and transportation planning. In this context, the task of developing a mathematical model of the motion of multilink road train (Figure 1) is very actual scientific and practical problem.

Figure 1. Heavy-duty trucks transporting OHCs

2 MATHEMATICAL EQUATIONS OF CURVILINEAR MOTION

This chapter of the paper describes the mathematical model of movement of a road train composed of truck and towed unit (trailer) on non-deformable support surface. It is known that longitudinal and transverse angles of the relative movement between the links of a road train when moving on a horizontal surface are not large [1, 2]. In this context movement of each separate link can be considered as movement of a rigid body on a horizontal plane with a rigid surface taking into account elevation angle in the direction of motion. The parameters of motion can be determined as a sum of translational motion of the center of mass and rotational movement around the center of mass (Figure 2).

Figure 2. Estimated traffic pattern of a train with a pivoted drawbar
The system of equations (1) describes this movement and allows calculating the current acceleration taking into account values of forces and moments acting on a train link.

\[
\begin{align*}
& a_x = \frac{dV_x}{dt} - \omega_y \cdot V_y = \frac{1}{m} \left( \sum_{i=1}^{s} R_{ix} - m \cdot g \cdot \sin(\alpha) + F_{x1} - P_{x1} \right), \\
& a_y = \frac{dV_y}{dt} + \omega_x \cdot V_x = \frac{1}{m} \left( \sum_{i=1}^{s} R_{iy} + F_{y1} - P_{y1} \right), \\
& J_z \cdot \frac{d\omega_z}{dt} = \sum_{i=1}^{s} M_{xix} + \sum_{i=1}^{s} M(R_i) + M_1, \\
& V_x = \frac{dx}{dt} = V_x \cdot \cos \theta - V_y \cdot \sin \theta, \\
& V_y = \frac{dy}{dt} = V_x \cdot \sin \theta + V_y \cdot \cos \theta, \\
& \omega_z = \frac{d\theta}{dt},
\end{align*}
\]

where \( m \) – the mass of the tractor; \( J_z \) – moment of inertia of the tractor with respect to the z axis; \( \vec{V} \) – the velocity vector of the center of mass of the tractor; \( \vec{a} \) – vector of acceleration of the center of mass of the tractor (absolute derivative of the velocity vector of the center of mass); \( \frac{d\vec{V}}{dt} \) – relative derivative of the velocity vector of the center of mass of the tractor; \( \vec{\omega} \) – angular velocity vector of the tractor entering a turn; \( \theta \) – angle of rotation of the tractor relative to the axis \( X' \); \( X', Y' \) – coordinates of the center of mass of the tractor in a fixed coordinate system; \( x, y \) – mobile system of coordinates associated with the tractor; \( \vec{R}_i \) – force vector form interaction with the ground acting on the i-th wheel; \( \vec{P}_w \) – vector of the force of air resistance; \( M_{xix} \) – moment of resistance of turning the i-th wheel; \( F_{x1} \) – the force exerted by the coupling device along the axis \( x \); \( F_{y1} \) – the force exerted by the coupling device along the axis \( y \); \( M_1 \) – moment transmitted to the body by the forces acting in the coupling device.

The system of equations describing the dynamics of the second link of the system (pivoted drawbar) can be written as:

\[
\begin{align*}
& a_x = \frac{dV_x}{dt} - \omega_z \cdot V_y = \frac{1}{m} \left( (F_{x21} + F_{x22}) \right), \\
& a_y = \frac{dV_y}{dt} + \omega_x \cdot V_x = \frac{1}{m} \left( (F_{y21} + F_{y22}) \right), \\
& J_z \cdot \frac{d\omega_z}{dt} = M_{21} + M_{22}, \\
& V_x = \frac{dx}{dt} = V_x \cdot \cos \theta - V_y \cdot \sin \theta, \\
& V_y = \frac{dy}{dt} = V_x \cdot \sin \theta + V_y \cdot \cos \theta, \\
& \omega_z = \frac{d\theta}{dt},
\end{align*}
\]

where \( F_{x21}, F_{x22} \) – forces acting on central link train from the front and rear coupling devices, respectively, along the axis \( X_{pr1} \); \( F_{y21}, F_{y22} \) – forces acting on central link train from the
front and rear coupling devices, respectively, along the axis $y_{pr1}$; $M_{21}, M_{22}$ – moments transmitted to the housing of central link of the road train, by the forces acting in the front and rear coupling devices, respectively.

The normal reactions of wheels $R_{zi}$ are redistributed as a result of the forces of air resistance, moments of wheel rolling resistance, acceleration of the center of mass and forces acting in coupling device.

Taking into account that the ends of vectors lie in the same plane it is possible to write:

$$
\begin{align*}
R_{11} + R_{22} + R_{33} + R_{43} + R_{55} + R_{66} + R_{77} + R_{88} &= m \cdot g + F_x, \\
R_{11} \cdot x_1 + R_{22} \cdot x_2 + R_{33} \cdot x_3 + R_{43} \cdot x_4 + R_{55} \cdot x_5 + R_{66} \cdot x_6 + R_{77} \cdot x_7 + R_{88} \cdot x_8 + \sum_{i=1}^{8} M_{fi} + P_{ui} \cdot Hz &= 0, \\
-x \cdot H_{wz} \cdot (g \cdot \sin (\alpha) + a_{z}) + F_{zi} \cdot H_{w} - F_{zi} \cdot d, \\
R_{11} \cdot y_1 + R_{22} \cdot y_2 + R_{33} \cdot y_3 + R_{43} \cdot y_4 + R_{55} \cdot y_5 + R_{66} \cdot y_6 + R_{77} \cdot y_7 + R_{88} \cdot y_8 + P_{zi} \cdot H_{w} &= -m \cdot a_{y} \cdot Hz + F_{zi} \cdot H_{w},
\end{align*}
$$

where $x_i, y_i$ – coordinates of the i-th wheel in the mobile coordinate system x-y; $H_z$ – the height of the center of mass of the vehicle; $H_{wx}, H_{wy}$ – the height of the points of application of forces of air resistance on the frontal and lateral views of the vehicle respectively; $H_{wp}$ – the height of the point of application of forces acting in the coupling device; $F_z$ – the vertical load on the coupling device.

The values of normal reactions $R_{zi}$ must be greater than zero. If one (or several) normal reaction turns negative, this reaction (reactions) is assumed to be zero, and the system is solved again (the number of equations reduces respectively). If when solving the system of equations four or more negative values of $R_{zi}$ are obtained, then the process of numerical simulation should be stopped because this mode corresponds to rollover case.

3 EXPERIMENTAL RESEARCH AND SIMULATION

As the object of the research a two-link vehicle was chosen. It consists of a tractor MAZ-7310 and a tow unit CMZAP-83881. General view of the object is shown on Figure 3. The parameters of movement were determined for two types of maneuvers: “Elk test” and “Going into corner”. Tests were conducted at maximum speed and at maximum steering angles. Registration of the trajectory was carried on basic points.

Figure 3. General view of the object of the research
In parallel the simulation was made on the basis of equations (1 – 3). The calculation scheme and simulation results of the test “Going into corner” are shown on Figure 4; the results of “Elk test” are shown on Figure 5.

![Figure 4. The scheme and simulation results of the test “Going into corner”](image)

The results obtained from the simulation of curvilinear motion on a smooth non-deformable surface have acceptable convergence with the experimental data for both “Elk test” and “Going into corner” types of tests. The divergence between control points of tractor and tow unit trajectories received in experiment and virtual tests was analyzed (Figure 6).

![Figure 5. The scheme and simulation results of “Elk test”](image)

The discrepancy on the basic parameters of curvilinear motion does not exceed 8…12%, it means that developed mathematical model of curvilinear motion is adequate and could be used for analysis of tractor and trailer trajectories and estimation of possibility of safety movement in real road infrastructure.
Figure 6. The comparison of trajectory coordinates received in real test and simulation (example of “Going into corner” test)

4 CONCLUSIONS

- A mathematical model of the movement of a road train is developed. This model allows predicting parameters of maneuverability during the transportation of OHC on a smooth rigid horizontal surface. The calculation should be performed at the stage of transportation planning. The peculiarity of the model consists in representation of a pivoted drawbar connecting the tractor with a trailer as a separate unit having a degree of freedom in the plane of movement in respect to two units of the road train.

- The developed model considers possibility of implementing various laws and algorithms of functioning of steering systems, various combinations of traction units of road trains and various features of their coupling devices.

- By comparing the results obtained during the simulation and field tests the adequacy of the developed model is proved. The model provides accuracy acceptable for predicting characteristics of curvilinear movement and can be applied for the needs of study of functioning of various laws of control of wheel turning with taking into account road conditions.

- The relative error on the basic parameters of curvilinear motion does not exceed 8% for the “Going into corner” maneuver and 12% – for “Elk test”.

5 ACKNOWLEDGEMENTS

The presented results were obtained in Nizhny Novgorod State Technical University n.a. R.E. Alekseev in the framework of the Federal Target Program «Research and development on priority directions of scientific-technological complex of Russia for 2014 - 2020 years» (agreement № 14.574.21.0107 (unique identifier of agreement - RFMEFI57414X0107)).

REFERENCES


Mathematical Model of All-Terrain Truck

Andrey V. Keller*, Vasiliy A. Gorelov†, Denis S. Vdovin†,

Pavel A. Taranenko#, Vladislav V. Anchukov#

*Department of Wheeled and tracked vehicles cars
South Ural state university
Lenina, 76, 454080 Chelyabinsk, Russia
andreikeller@rambler.ru

†Department of Wheeled vehicles
Bauman Moscow state technical university
ul. Baumanskaya 2-ya, 5, 105005 Moscow, Russia
gvas@mail.ru
vdovin@kamaz.ru

#Department of Applied Mechanics, Dynamics and Strength of Machines
South Ural state university
Lenina, 76, 454080 Chelyabinsk, Russia
pataranenko@gmail.com
vv.anchoukov@gmail.com

ABSTRACT

To research all-wheel-drive 6x6 terrain truck dynamics the decomposition method was used. According to this method the model was divided into two subsystems: 1) Engine, driveline, driving wheels; 2) Chassis, suspension, driving wheels and road surface. To make mathematical model special simulation software Siemens LMS Imagine.Lab and Matlab/Simulink were applied. As a result we have a model that allows us to evaluate the external vehicle dynamics and internal driveline dynamics. After the application of the special simulation software we have the following results: 1) When using of locking inter-axle differential, drive axles of the vehicle are overloaded by double and triple dynamic forces against static forces. It can be calculated with maximum engine torque in consideration of static traction forces and static reactive toques on drive axles; 2) In case of wheels slipping, when using locking interwheel differential, dynamic redistribution of torques influences the dynamic conditions of driving wheels interaction and road surface and depends on vertical forces dynamic redistribution in contact with road surface.

Keywords: vehicle dynamics; mathematical modeling; power distribution.

1 INTRODUCTION

Trucks are one of the most widely spread kinds vehicles used in different road conditions. To make the functioning of trucks in the conditions with the low traction coefficient more effective it is necessary the further development of the power distribution system.

Simulation modeling of the vehicle dynamics is widely spread in the world scientific practice. The main reason is to get the necessary features of the developed item without any experiments on the stage of designing that influences positively on the quality of the launched products and makes the cost of price lower.

Now days there are a variety of approaches to vehicle dynamics modeling, but not all of them are universal. Here are some of them.

In the papers [1, 3] some mathematical models of cars are designed. The model of SUV with active steering and interaxial power dividing unit is suggested in [1]. Mathematical model proposed in [3] shows the interconnection of dynamic processes and the function of powertrain unit. In the research work [2] the mathematical model of heavy-duty truck is developed. Vehicle dynamics is evaluated by means of inverse dynamics method. Active differential and traction control system algorithms are suggested. In the article [4] the vehicle dynamics model with active steering and individual front wheels braking is depicted. The fuzzy-logic method is applied for vehicle steering and getting the necessary trajectory. The authors of the
research work [5] developed the mathematical model of vehicle dynamics with active differential and control system. There was an estimation of vehicle drivability and powertrain dynamics. In [6] the mathematical model of the bench test vehicle simulation is shown, allowing to simulate the series of standard bench tests. The paper [11] is devoted to principals and methods of torque distribution in the operation of anti-slipping system.

The effective vehicle power distribution is the main factor defining vehicle movability. In the world science there is a great interest to this question. In the scientific works [7, 8, 12] the principals and methods of power distribution are proposed giving the possibility to solve the problem of providing the necessary achievable degree of terrain truck movability. The authors developed the new methodology of defining the power regularity between driving wheels and their adaptation to real driving condition.

In [9] the model of autonomously controlled mechanism of power distribution is suggested. And there is also a unit control algorithm.

In the paper [10] the mathematical models of four active differential types are presented. There is ALSD, clutch TVD superposition, stationary clutch TVD, 4WD TVD. The developed mathematical models were verified experimentally by the authors and findings were approved by the high model accuracy.

Analyzing the works depicted above we can make a conclusion that nowadays there are still some unsolved problems, connecting with the vehicle dynamics estimation considering dynamic processes in all the main vehicle subsystems: engine, transmission, suspension and wheels. Therefore the designing of correct and detailed model is the necessary point of vehicle dynamics evaluation.

2 ALL-TERRAIN TRUCK MODEL

For the correct description of vehicle movability it is necessary to pay attention to the dynamics of all the systems, for this decomposition method is applied. Therefore we divide the vehicle into several subsystems:

- Engine, transmission, driving wheels;
- Car body, suspension, driving wheels and supporting ground surface.

Hence there is a necessity in application of mathematical analysis for dynamic system description. Imagine the process of vehicle dynamics as rigid multibody system and define the main forces, influencing the mechanical system (Figure 1).

![Figure 1. Truck scheme in straight line motion (G – truck center of gravity (COG); XYZ – axes of global coordinate system; P_x, P_y – axes of local coordinate system with origin in vehicle COG; R_{x,y} – wheel longitudinal force; R_{y,z} – wheel lateral force; R_{z,y} – wheel normal force; c_{d,y} – wheel damping; k_{d,y} – wheel stiffness; c_p – suspension damping; k_p – suspension stiffness; P_w – aerodynamic resistance force).](image)
The wheel interaction with the road surface with enough accuracy is defined by Pacejka Magic formula. According to this the longitudinal, lateral and aligning moment are calculated in the following way.

Longitudinal force:

\[ F_x = D \sin(C \cdot \arctan\{B(1 - E)(\sigma + S_h) + E \cdot \arctan[B(\sigma + S_h)]\}) + S_v \]  \hspace{1cm} (1)

Lateral force:

\[ F_y = D \sin(C \cdot \arctan\{B(1 - E)(\alpha + S_h) + E \cdot \arctan[B(\alpha + S_h)]\}) + S_v \]  \hspace{1cm} (2)

Aligning moment:

\[ M_z = D \sin(C \cdot \arctan\{B(1 - E)(\alpha + S_h) + E \cdot \arctan[B(\alpha + S_h)]\}) + S_v \]  \hspace{1cm} (3)

where

\( \sigma \) - longitudinal slip;
\( \alpha \) - slip angle;

Coefficients \( B, C, D, E \) and sums \( S_h, S_v \) are defined according to tire parameters getting experimentally.

Paying attention to obtaining coefficients, the graphics of longitudinal, lateral forces and aligning moment dependence as slip function will get the views as follows (Figure 2).

\[ \text{Figure 2. Truck tire forces.} \]

2.1 Equations of motion

For description of transmission dynamics we use Lagrange equations of the II kind with undefined multipliers.

\[ \frac{d}{dt}\left( \frac{\partial T}{\partial \dot{q}_i} \right) - \frac{\partial T}{\partial q_i} = Q_{q_i} + \sum_{s=1}^{n} \mu_s \cdot a_{si} + \sum_{j=1}^{m} \xi_j \frac{\partial f_j}{\partial q_i}. \]  \hspace{1cm} (4)

where

\( T \) – kinetic energy;
\( q_i \) – generalized coordinates;
\( Q_{q_i} \) – generalized forces;
\( n \) – number of nonholonomic constraints;
\( \mu_s \) - Lagrange undetermined multipliers;
\( a_{si} \) – generalized coordinates multipliers in nonholonomic constraints equations;
\[
\sum_{i=1}^{l} a_{si}(q, t) \cdot \dot{q}_i + b_s(q, t) = 0; \tag{5}
\]

\(\zeta_i\) – constraint equations multipliers \(f_j\) for \(m\) redundant coordinates of the system.

The mathematical model of driveline dynamics is described in the following second order differential equations.

\[
J_e \ddot{\phi}_e = M_e [\dot{\phi}_e, h(t)] - M_c (\dot{\phi}_e, \dot{\phi}_c, t) + M_{res}^e = 0, \tag{6}
\]

\[
J_{cl} \ddot{\phi}_{cl} = c_{cl} (\phi_{cl} - \phi_{gb1}) + b_{cl} (\phi_{cl} - \phi_{gb1}) = M_{cl} (\phi_e, \phi_{cl}, t), \tag{7}
\]

\[
J_{gb1} \ddot{\phi}_{gb1} - c_{gb1} (\phi_{cl} - \phi_{gb1}) - b_{gb1} (\phi_{cl} - \phi_{gb1}) + c_{gb1} (\phi_{gb1} - \phi_{gb2} - (i_{gb1} - 1) \phi_{gb2}^r) + b_{gb1} (\phi_{gb1} - i_{gb1} \phi_{gb2} - (i_{gb1} - 1) \phi_{gb2}^r) = -M_{res}^{gb1}, \tag{8}
\]

\[
J_{gb2} \ddot{\phi}_{gb2} - c_{gb2} (\phi_{gb1} - \phi_{gb2} - (i_{gb1} - 1) \phi_{gb2}^r) - b_{gb2} (\phi_{gb1} - \phi_{gb2} - (i_{gb1} - 1) \phi_{gb2}^r) + c_{gb2} (\phi_{gb1} - \phi_{gb2} - (i_{gb1} - 1) \phi_{gb2}^r) + b_{gb2} (\phi_{gb1} - i_{gb1} \phi_{gb2} - (i_{gb1} - 1) \phi_{gb2}^r) = -i_{gb1} M_{res}^{gb2}, \tag{9}
\]

\[
J_c \ddot{\phi}_c - b_{gb1} (\phi_{gb1} - \phi_{gb2} - (i_{gb1} - 1) \phi_{gb2}^r) + b_{gb2} (\phi_{gb1} - \phi_{gb2} - (i_{gb1} - 1) \phi_{gb2}^r) + c_c [\phi_{gb1} - \phi_{gb2} - (i_{gb1} - 1) \phi_{gb2}^r] = M_{res}^c \[\phi_e, h(t)\]. \tag{10}
\]

\[
J_e \ddot{\phi}_e - i_e c_e [\phi_{gb2} - \phi_{gb3}] + b_e (\phi_{gb2} - \phi_{gb3}) + c_{cl} (\phi_{gb1} - \phi_{gb2} - (i_{gb1} - 1) \phi_{gb2}^r) + b_{cl} (\phi_{gb1} - \phi_{gb2} - (i_{gb1} - 1) \phi_{gb2}^r) = \dot{M}_{res}^e, \tag{11}
\]

\[
J_{gb1} \ddot{\phi}_{gb1} - c_{gb1} (\phi_{gb1} - \phi_{gb2} - p \phi_{gb2} - (1 + p) \phi_{gb1}) - b_{gb1} (\phi_{gb1} - \phi_{gb2} - p \phi_{gb2} - (1 + p) \phi_{gb1}) - M_{res}^{gb1} = 0, \tag{12}
\]

\[
J_{gb2} \ddot{\phi}_{gb2} - c_{gb2} (\phi_{gb1} - \phi_{gb2} - p \phi_{gb2} - (1 + p) \phi_{gb1}) - b_{gb2} (\phi_{gb1} - \phi_{gb2} - p \phi_{gb2} - (1 + p) \phi_{gb1}) = -M_{res}^{gb2}, \tag{13}
\]
\[
J_{bg3} \dot{\phi}_{bg3} - c_{ds3} (\phi_{bg2} - \phi_{bg3}) - b_{ds3} (\phi_{bg2} - \phi_{bg3}) \\
+ \left[ c_{wd3} \left( \frac{\phi_{bg3}}{i_0} - \phi_{w3} \right) + b_{wd3} \left( \frac{\phi_{bg3}}{i_0} - \phi_{w3} \right) \right] / i_0 = -M_{ds3}^{res},
\]

Nomenclature is presented in appendix A.

The parameters of vehicle movability in steady fixed coordinate system \( OXYZ \) by solving Lagrange equations of the II kind are depicted below.

\[
\begin{cases}
\dot{\omega} = T_1 - G_s h_{COG} \lambda \left( \dot{V}_x - V_y \omega \right) / J_z, \\
\dot{V}_x = V_y \omega + T_2, \\
\dot{V}_y = V_x \omega + T_3,
\end{cases}
\]

\[
T_1 = \sum_{i=1}^{2} R_y^{i} a + \sum_{i=1}^{2} R_y^{i} b_2 + \sum_{i=1}^{2} R_y^{i} b_3 + 0.5 \left( R_{x1}^{right} + R_{x1}^{left} \right) B_1 \\
+ 0.5 \left( \sum_{j=2}^{3} R_{xj}^{right} + \sum_{j=2}^{3} R_{xj}^{left} \right) B_2 / J_z,
\]

\[
T_2 = \sum_{j=1}^{3} \sum_{i=1}^{2} R_{xj}^{i} / G
\]

\[
T_3 = \sum_{j=1}^{3} \sum_{i=1}^{2} R_{yj}^{i} / G
\]

Nomenclature for these equations is presented in appendix A.

2.2 Model realisation

As seen from aforementioned equations the analytical description of vehicle dynamics processes is very difficult. Therefore there is the necessity to find out alternative methods for vehicle exploiting qualities evaluation. One of these methods is the simulation modeling. The main advantages of simulation modeling are the possibility of solving the complex non-linear mathematical problems with high accuracy as well as finding specific technical decisions while designing the item.

For the vehicle dynamics modeling we use CAE LMS Imagine. Lab AMESim and MATLAB/Simulink, which were successfully applied in this definite field of research. LMS Imagine.Lab AMESim is widely used in the automotive industry for solving wide range of problems.

Block structure is applied in this CAE, all the systems are comprised from elements describing the function of a certain mechanism, electric unit, logic element etc.

The common view of the model is depicted in the figure 3. This model includes all the aforementioned subsystems: engine, driveline, suspension, car body and wheels with tire model.
Car body is a rigid one influenced by powertrain units, environment and driving conditions.

The powertrain unit comprises engine, functioning according to throttle position (Fig. 5) and detailed driveline model (Fig. 4) includes clutch, ideal CVT, that is necessary for realizing the gear ratio, compliant driving shafts with cardan joints, five differentials (two inter-axial and two interwheeled) and bevel gears on each axle.

Suspension considers drive connection and the dynamics working process of the solid axle distributing force and vibrations to the body. The interaction of the wheeled running gear and area of bearing is realized with the help of built-in models of Pacejka 1996.

Figure 3. The common view of the model.

Figure 4. Mathematical model of the driveline (1 – clutch; 2 – ideal CVT; 3 – drive shaft; 4 – inter-axial)
differential; 5 – bevel gear; 6 – interwheeled differential).

**Figure 5.** Diesel engine characteristic.

The “virtual driver” system sustains the mode of driving and comprises the set of probing vehicle technologies and logic elements. The “virtual driver” solves the following problems: throttle pedal position and clutch control, speed control, braking and driving trajectory.

### 2.3 Simulation results

With the help of the realized model we were succeeded to get a loading process of inter-axial blocking differential actuation. So we consider the following controlling case.

The vehicle starts driving with the first speed up in the hill and continues driving without changing the speed. Driving up in the hill is accompanied with the road coating changes.

<table>
<thead>
<tr>
<th>Driving process</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Initial vehicle state</strong></td>
</tr>
<tr>
<td><strong>Angle of raising, grad</strong></td>
</tr>
<tr>
<td><strong>Friction coefficient on the main surface</strong></td>
</tr>
<tr>
<td><strong>Slipping traction coefficient (length of surface 5 m, starting with 10 m)</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Driveline mode</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Differential conditions</strong></td>
</tr>
<tr>
<td><strong>Speed number</strong></td>
</tr>
<tr>
<td><strong>Speed control</strong></td>
</tr>
<tr>
<td><strong>Splitting the torque between front axle and the truck</strong></td>
</tr>
<tr>
<td><strong>Throttle position</strong></td>
</tr>
</tbody>
</table>

In this driving mode the driveshaft loading was obtained and therefore influencing the half-coupling of the inter-axial differential in the gearbox. Simulation results depicted below (Fig. 6; Fig. 7).
According to the taken loads the FEM (finite element model) of the half-coupling differential locking was developed and the active stresses at the starting moment was estimated (Fig. 8-9).

Figure 6. Rotating speed changes in the driving process, rpm (1 – rotating engine speed, 2 – output spindle rotating speed of gearbox to the front axle, 3 - output spindle rotating speed of gearbox to the rear axles).

Figure 7. Torques driving changes, Nm (1 – engine torque, 2 – output drive shaft torque of transfer case to the front axle, 3 – output drive shaft torque of transfer case to the rear axles).

Figure 8. Received stresses for first half-coupling differential locking 1, Pa.
The obtained stress values turned out to be very high. Therefore to safe functionality of differential locking coupling it is necessary to take steps for making loading at the starting moment a little lower. Such steps could be: geometric changes of gear profile, balancing the driveshaft speed with the help of braking action towards the slipping wheels, decreasing the rotating moment with the help of fuel control.

3 CONCLUSIONS

The achieved mathematical model was applied for researching the different kinds of driveline vehicle dynamics including the evaluation of differential loading mechanisms and developing their controlling algorithms.

ACKNOWLEDGMENT

The work was performed under financial support of the ministry of education and science of Russian Federation in the frames of the project generating scientific-and-technical solutions on controlling power distribution in the cargo trucks transmission trains for increasing their power and fuel efficiency according to an agreement № 14.574.21.0106 DATED 09.08.2014.

Unique identifier of the applied scientific research (project) RFMEFI57414X0106.

Appendix A

Nomenclature for eq. 6-14 depicted below:

- \( J_e \) - Engine moment of inertia;
- \( \phi_e \) - Engine rotation angle;
- \( M_e \) - Engine torque;
- \( h(t) \) - Throttle position;
- \( M_{cl} \) - Clutch torque;
- \( \phi_{cl} \) - Clutch rotation angle;
- \( M'_{res} \) - Engine resistant torque;
- \( J_{cl} \) - Clutch moment of inertia;
- \( c_{cl} \) - Clutch stiffness;
- \( b_{cl} \) - Clutch damping ratio;
- \( \phi_{gb1} \) - Primary gearbox shaft rotation angle;
\[ J_{gb1} \] - Primary gearbox shaft moment of inertia;
\[ c_{gb} \] - Gearbox stiffness;
\[ b_{gb} \] - Gearbox damping ratio;
\[ i_{gb} \] - Gearbox gear ratio;
\[ \phi_{gb2} \] - Secondary gearbox shaft rotation angle;
\[ \phi_e^R \] - "Reactive" engine rotation angle;
\[ J_{gb2} \] - Secondary gearbox shaft moment of inertia;
\[ c_{tc} \] - Transfer case stiffness;
\[ b_{tc} \] - Transfer case damping ratio;
\[ \phi_{tc} \] - Transfer case output shaft rotation angle;
\[ i_{tc} \] - Transfer case gear ratio;
\[ M_{gb}^{res} \] - Gearbox resistant torque;
\[ J_e^R \] - "Reactive" engine moment of inertia;
\[ c_e^R \] - "Reactive" engine stiffness;
\[ b_e^R \] - "Reactive" engine damping ratio;
\[ c_{ds1} \] - Front axle drive shaft stiffness;
\[ p \] - inter-axial differential gear ratio;
\[ k_{bias} \] - BIAS ratio;
\[ c_{ds2} \] - Middle axle drive shaft stiffness;
\[ \phi_{bg1} \] - Front bevel gear rotation angle;
\[ \phi_{bg2} \] - Middle bevel gear rotation angle;
\[ M_{tc}^{res} \] - Transfer case resistant torque;
\[ J_{bg1} \] - Front bevel gear moment of inertia;
\[ c_{z} \] - Bevel gear mechanism stiffness;
\[ b_{z} \] - Bevel gear mechanism damping ratio;
\[ c_{wd1} \] - Front axle wheel drive shaft stiffness;
\[ b_{wd1} \] - Front axle wheel drive shaft damping ratio;
\[ i_0 \] - interwheeled differential gear ratio;
\[ \phi_{w1} \] - Front wheels rotation angle;
$M_{ds_1}^{res}$ - Front axle wheel drive shaft resistant torque;

$J_{bg_2}$ - Middle bevel gear moment of inertia;

c_{ds_3} - Rear axle drive shaft stiffness;

$\phi_{bg_3}$ - Rear bevel gear rotation angle;

$b_{ds_3}$ - Rear axle drive shaft damping ratio;

$b_{ds_2}$ - Middle axle drive shaft damping ratio;

$b_{ds_1}$ - Front axle drive shaft damping ratio;

$c_{wds_2}$ - Middle axle wheel drive shaft stiffness;

$b_{wds_2}$ - Middle axle wheel drive shaft damping ratio;

$\phi_{w_2}$ - Middle wheels rotation angle;

$M_{wds_2}^{res}$ - Middle axle wheel drive shaft resistant torque;

$J_{bg_3}$ - Rear bevel gear moment of inertia;

$c_{wds_3}$ - Rear axle wheel drive shaft stiffness;

$b_{wds_3}$ - Rear axle wheel drive shaft damping ratio;

$\phi_{w_3}$ - Rear wheels rotation angle;

$M_{wds_3}^{res}$ - Rear axle wheel drive shaft resistant torque.

Nomenclature for eq. 15-18 presented below:

$\omega$ - Vehicle body rotary velocity;

$G_s$ - Total sprung mass;

$G$ - Vehicle mass;

$h_{COG}$ - Center of gravity height;

$\lambda$ - Roll angle;

$V_x$ - Velocity in longitudinal direction;

$V_y$ - Velocity in lateral direction;

$J_z$ - Vehicle body moment of inertia;

$R_x$ - Longitudinal wheel reaction;

$R_y$ - Lateral wheel reaction;

$a$ - Distance between vehicle front axle and center of gravity;

$b_2$ - Distance between vehicle center of gravity and middle axle;
$b_3$ - Distance between vehicle center of gravity and rear axle;

$B_1$ - Front track width;

$B_2$ - Rear track width.

REFERENCES


Multibody and Finite Element models of a leaf-spring suspension for vehicle dynamics applications: numerical model, tests and correlation

A.Toso*, U.Facchin#, S.Melzi†

* Siemens Industry Software NV
Interleuvenlaan 68, B-3001 Leuven, Belgium
alessandro.toso@siemens.com

# Department of Mechanical Engineering
Politecnico di Milano
Via La Masa 1, 20156 Milan, Italy
umberto.facchin@mail.polimi.it

† Department of Mechanical Engineering
Politecnico di Milano
Via La Masa 1, 20156, Milan, Italy
stefano.melzi@polimi.it

ABSTRACT
Leaf springs are still used as components of suspension systems especially in trains, trucks and heavy duty vehicles. Mathematical models of leaf springs are therefore needed for the prediction of comfort and handling of vehicles at design stage. This paper is focused on the Multibody and Finite Element approaches with the goal of finding a good compromise between accuracy and computational speed. A leaf spring, part of the rear suspension of a commercial pick-up truck, has been tested to determine its response both in static and dynamic conditions (harmonic loading at different frequency up to 10 Hz and with different preload levels). Finite element and multibody models were developed; their outputs were compared with experimentation results and their performance was analysed terms of accuracy and computational efficiency.

Keywords: Leaf Spring, Finite-element model, Multibody model, Experimental test.

1 INTRODUCTION
A leaf spring consists of one or more slender curved blades packed into an assembly, bolted in the middle part (Figure 1) whose extremities are in general connected to the vehicle frame by means of rubber bushings. During the application of a load, each leaf is in contact with one or two other leaves providing in this way a mechanism for load transfer and dissipation of energy. The presence of an additional stage, as shown in Figure 1, provides a dramatic change in stiffness when its leaves are actively in contact with the primary spring.

Figure 1. Variable rate leaf spring.

Though representing a rather old technical solution (first examples date back to the middle age), leaf springs are still used as components of suspension systems especially in trains, trucks and heavy duty vehicles. Three main advantages of leaf spring suspensions over other layouts are: a) simplicity: the high rigidity allows longitudinal and lateral location of the axle without using additional linkages; b) better load distribution: suspension load is distributed over the vehicle’s
chassis, while when using coil springs it concentrates on a small area; c) variable stiffness rate: through the use of more stages, spring stiffness can be varied keeping a simple assembly. Despite the simplicity of the component, the modelling of a leaf spring still represents a challenging task. Technical literature offers a number of analytical [2][3], semi-empirical [1], non-physical [4], finite element [5][6][7] and multi-body models [8][9][10] developed for different purposes: quasi-static characterization, fatigue life estimation, influence on vehicle’s handling and comfort. Models are characterized by different degrees of complexity, accuracy and computational effort. When it comes to the analysis of the effect on vehicle comfort, handling and stability both accuracy and computational efficiency are to be considered; in addition two aspects should be properly addressed [1]: a) contacts among blades which lead to different spring rates and to friction; b) inertial effects. Contacts among blades are not always gradual so that spring stiffness displays a step-like pattern. Friction constitutes the main dissipating mechanism of a leaf spring resulting in a very small effect of excitation frequency on dissipated energy. Weight is one of the main drawbacks of this technical solution: as a consequence, inertial effects deeply affect the response of a leaf spring for frequencies higher than 3-5 Hz. This means that the dynamic response can be remarkably different from the quasi-static one [11]. This paper is focused on the Multibody and Finite Element approaches with the goal of finding a good compromise between accuracy and computational speed bearing in mind the final target of the modelling activities that is the vehicle dynamics performance evaluation for handling and ride comfort. The spring under examination is part of the rear suspension of the pick-up truck Ford F350 Super Duty. It has been tested in the laboratories of the Mechanical Engineering Department of Politecnico di Milano both in static and dynamic conditions (harmonic loading at different frequency up to 10 Hz and with different preload levels). Moreover, an experimental modal analysis was carried out to identify eigenfrequencies, eigenmodes and modal damping of the component. The multibody model was built in LMS Virtual.Lab Motion [3] using a discretization of each leaf in rigid bodies connected by linear elastic beams. The contact was modelled as a simplified sphere-to-extrude contact that has the advantage of being computationally efficient and it’s suited for this type of application where the contact areas are known. The paper is organized as follows: as first a brief description of the experimental activity and its main results is reported. Then finite-element and multi-body models are presented, their outputs are compared with experimentation results and their performance is analysed terms of accuracy and computational efficiency.

2 EXPERIMENTAL CHARACTERIZATION

2.1 Static and dynamic response
The response of the leaf spring has been characterized using the test apparatus shown in Figure 2. The extremities of the leaf spring are linked by pivots to two blocks (sliding blocks) moving along a horizontal straight slideway. Sliding blocks allow the leaf spring extremities to move along the longitudinal direction. The hydraulic actuator controls the motion along the vertical direction of the spring centre; it can develop a maximum force of 15 kN with a maximum stroke of 254 mm. An interface ring made up of aluminium is present between the load cell and the leaf spring; this component allows to distribute the force over a larger area. The hydraulic actuator is provided with a load cell and an LVDT whose signals are recorded during tests to characterize the spring response in terms of force-displacement curve. The load cell guarantees at least 99% accuracy over the force range while the maximum error in the estimation of piston position is within 0.35%. The measurement set up is completed by an optical laser sensor used to measure the transversal displacement of one of the two sliding blocks. The displacement of the second block is assumed to be equal to the first one. The optical sensor has a range of ±25 mm with a resolution of 20 μm.
The leaf spring response has been investigated considering both quasi-static and dynamic load conditions.

The quasi-static test was performed controlling the force of the piston. A static pre-load of 1 kN was applied at the beginning then the actuator force followed a triangular wave signal reaching a maximum value of 14 kN. The period of the wave was 60 seconds. The relation between the position of the actuator head and the force measured by the load cell is displayed in Figure 3. The picture clearly put into evidence the change in the slope of the force/compression relation due to the contribution of the third leaf of the spring. This occurs when the position of the piston is around -45 mm, corresponding to a compression of the leaf spring of nearly 147 mm. The exam of reveals a small dissipation is present also in quasi-static condition. Dissipations are mainly due to friction between the leaves. The energy dissipated in a single load cycle is around 35 J, while the maximum energy stored in the spring is approximately 1120 J.

Also a series of dynamic tests was carried out according to the conditions listed in Table 1. Tests were performed considering four levels of static preload and different excitation frequencies from 0.1 to 10 Hz. This time the position of the actuator was controlled to obtain load cycles with an amplitude of 20 mm.

In each dynamic test the piston followed a sinusoidal wave with given amplitude. As an example Figure 4 shows the load cycle measured during the 1-Hz test with a pre-load of 4 Hz. Data were low-pass filtered to remove harmonic components above 20 Hz. The figure reports also the load cycle removing inertial effects. These last are associated with the oscillating motion of sliding blocks, interface ring and also of the leaf spring itself. According to technical literature [3], 70%-90% of the mass of a leaf spring the spring takes part to the oscillating motion. Inertial effects were cancelled considering 80% the mass of leaf spring and 100% the mass of the other components.

Considering data reported in Figure 4, inertial effects are completely negligible at 1 Hz. This is no longer true when data relevant to the test at 5 Hz are considered. Figure 5 shows that inertial effects are able to significantly reduce the slope of the load cycle. If the test frequency is increased above 7 Hz, the slope of the load cycle will become negative. Removing inertial effects allows to determine the shape of the load cycle associated only with spring stiffness and dissipation.

Altogether, removing inertial effects, dynamic tests showed that: a) the shape of load cycles is not elliptical and is characterized by sharp edges; b) the shape and the slope of the load cycle is minimally influenced by frequency. Both the aspects suggest a dissipation mechanism associated with internal friction.

<table>
<thead>
<tr>
<th>Pre-load [kN]</th>
<th>Frequency [Hz]</th>
<th>Displacement [mm]</th>
</tr>
</thead>
<tbody>
<tr>
<td>4,7,9,12</td>
<td>0.1, 0.25, 0.5, 0.75, 1, 1.5, 2, 3, 4, 5, 6, 7, 8, 9, 10</td>
<td>±10 mm</td>
</tr>
</tbody>
</table>
2.2 Modal Analysis

The experimental modal analysis (rowing hammer test) was carried out in order to completely characterize each blade in terms of natural frequencies, mode shapes and modal damping. Siemens provided the instrumentation for the tests. The test setup is shown in Figure 6: the 1D accelerometer used was carefully placed in an anti-node and two soft elastics were adopted to better reproduce the free-free condition. Finally, acquisition and processing of the signal were performed with the software LMS Test.Lab in conjunction with the LMS SCADAS front-end. An example of frequency response function is shown in Figure 7, while Table 2 lists the eigenfrequencies and modal damping. The first peak below 3 Hz, is relevant to a rigid mode introduced by the elastics that hang the blades.

For what concerns the modal damping the results indicate that the first and second blades have lower damping values if compared to the helper leaf.

| Mode | Master blade | | Second blade | | Helper leaf |
|------|--------------|----------------|--------------|----------------|
|      | Freq. [Hz]   | Damp. [%]      | Freq. [Hz]   | Damp. [%]      | Freq. [Hz]   | Damp. [%]      |
| 1    | 16.9         | 1.4284         | 30.4         | 0.41           | 140.4        | 0.5755         |
| 2    | 50.7         | 0.0953         | 89.5         | 0.2610         | 335.7        | 0.4317         |
| 3    | 103.3        | 0.0043         | 176.4        | 0.0391         | /            | /              |
| 4    | 174.9        | 0.0129         | 294.0        | 0.0431         | /            | /              |
| 5    | 263.7        | 0.0316         | 439.5        | 0.0092         | /            | /              |
| 6    | 371.2        | 0.4216         | /            | /              | /            | /              |
| 7    | 491.4        | 0.0963         | /            | /              | /            | /              |

Table 2. Modal analysis results (0-500 Hz)
3 FINITE ELEMENT MODEL

The finite element analysis was performed using SAMCEF Mecano [13], the finite element solver for non-linear analysis developed by Siemens PLM Software. The three blades (Figure 8) were created with an external software and then imported into the SAMCEF environment where material properties, kinematic and dynamic constraints, and elements type would have been defined.

The viscoelastic material was preferred because of its ability to represent both the viscous and elastic behaviours, in particular the Kelvin Voigt viscoelastic model was chosen. The Kelvin Voigt model describes the behaviour as hookean stiffness and newtonian structural damping in parallel. The strain-deformation law is:

$$\sigma(t) = E \varepsilon(t) + \alpha E \dot{\varepsilon}(t)$$  \hspace{1cm} (1)

Therefore, since $E$ is fixed according to the stiffness of the system, the parameter $\alpha$ is responsible for introducing damping into the structure. The parameters density, Young’s modulus, and Poisson’s ratio were assigned according with the standard material. For what concerns the elements type the presence of eyehookes led to prefer the tetrahedral elements since they are able to adapt to any shape, in particular the second order type was chosen in order to correctly represent the bending.

The contact was defined using the uncoupled variable penalty method (schematically described in Figure 9) that requires more computational time if compared to coupled methods but on the other hand enhances convergence in analysis with friction. The penalty method relaxes the condition of infinitely rigid contact by adding a penalty function: small penetrations are allowed and the contact pressure is related to such interferences through a penalty function. Basically it can be seen as a spring that works only in compression: when the $AB$ length becomes lower then $B_{\text{min}}$ the contact is active and the penalty factor $K_n$ defines the contact stiffness. The contacts were defined only between the parts involved, thus patches on the superior part of the master blade were created in order to reduce the active nodes in contact. Finally two simple ideal ground-slider elements were used to connect the ground with eyehooks.

To validate the model the correlation with quasi-static and dynamic tests data was analysed. Both the static and dynamic simulations required the clamping procedure, since the blades were created in their free condition.

3.1 Simulation of the quasi-static test

To reach the desired stiffness no tuning was needed but an optimization process on the size of the tetrahedral elements. The elements size was finally reduced to 15 mm and the stiffness results the 3.66% higher than the actual one.
3.2 Simulations of dynamic tests

The correlation with dynamic data was carried out on the basis of the 4000 N preload case in two steps: firstly the reliability to get the correct stiffness and inertia forces at high frequencies was proved by matching filtered tests curves and over damped simulations with a proper coefficient damping $\alpha$ per each frequency, then the capability to represent the correct resonances was tested by using a unique damping coefficient allowing the generation of resonances. The first approach gave excellent results: Figure 11 and Figure 12 show tests and simulations overlapping and Table 3 lists the values of the coefficient $\alpha$ used for each simulations. Therefore, by varying the damping coefficient, it is possible to control the energy dissipation while matching the stiffness of the system. The only considerable difference is visible in the 8 Hz test: the non-linearity introduced by the piston in the direction change cannot be reproduced since the simulations are performed with an ideal connection piston-leaf spring.

Table 3. Damping coefficient $\alpha$ vs frequency.

<table>
<thead>
<tr>
<th>Frequency [Hz]</th>
<th>0.25</th>
<th>1</th>
<th>3</th>
<th>5</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha$</td>
<td>0.05</td>
<td>0.01</td>
<td>0.0035</td>
<td>0.0025</td>
<td>0.0015</td>
</tr>
</tbody>
</table>

The second approach, on the other hand, considered the use of a unique low damping coefficient assuming an energy dissipation mechanism mainly associated with a higher friction coefficient. This approach actually represents an operating condition closer to real one and the resulting solution is embeddable in a larger vehicle model.

The results shown in Figure 13 were obtained by using a damping coefficient of $1 \times 10^{-5}$ s. The FFT reported in Figure 13 highlights the presence of the higher harmonic components, in particular the one at 70 Hz, that mostly affect the results.
Figure 13. Dynamic load cycles and FFT at 3 and 5 Hz: correlations between tests and FEM simulations

The modal analysis in Figure 14 was then performed on the pre-stressed structure. The presence of the eigenvalue at 72 Hz is consistent with the spectrum analysis of the simulations and tests.

![Figure 14. First four numerical mode shapes](image)

3.3 Computational time

The refined discretization necessary to correctly represent the leaf spring stiffness weighs considerably on computational time. The static simulation requires 27 minutes to calculate two quasi-static cycles (120 seconds). The dynamic simulations stress the CPU-time: to compute 5 seconds of simulation 1 hour and half is required with the first approach, while 3-4 days are necessary with the second one, depending on the frequency.

4 MULTIBODY MODEL

The multibody analysis was performed using LMS Virtual.Lab Motion, a software developed by Siemens PLM Software. The model was created by means of the graphical user interface LMS Leaf Spring Tool. It assembles the leaf spring in its preloaded clamped positions. The elastic force is provided by elastic beam elements that connect rigid bodies. The beam element indeed produces forces along and torques about the three axes of the element when the relative position between two bodies changes. During the simulation relative displacements and velocities are calculated between the two bodies and forces are thus obtained by the equation:

\[ f = [K]x + [D] \dot{x} \]  \hspace{1cm} (3)

The stiffness matrix \( K \) is calculated using a simple linear beam theory [14]:
\[
K = \begin{bmatrix}
\frac{EA}{l} & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{12EI_y}{l^3} & 0 & 0 & 0 & \frac{-6EI_y}{l^2} \\
0 & 0 & \frac{12EI_z}{l^3} & 0 & \frac{6EI_z}{l^2} & 0 \\
0 & 0 & 0 & \frac{GI_y}{l} & 0 & 0 \\
0 & 0 & 0 & 0 & \frac{4EI_z}{l} & 0 \\
0 & \frac{-6EI_y}{l^2} & 0 & 0 & 0 & \frac{4EI_z}{l}
\end{bmatrix}
\]  

(4)

Where \( A \) is the surface, \( l \) the free length, \( E \) the Young’s Modulus, \( G \) the shear modulus, and \( I \) the moments of inertia respectively in x, y, and z.

For what concerns damping, the structural model was preferred. It is defined as the product between a damping factor \( c \) and the diagonal of the stiffness matrix. It can be related to the critical damping ratio by:

\[
[D = c[K] \\
D = 2\xi\sqrt{[K]M}]
\]  

(5)

Finally to reach the final configuration the shims, rail, sliders, and bushing force elements (connection sliders-eyehooks) were added. The correlation between tests and MBS was carried out according to the FEM case.

4.1 Simulation of quasi-static tests

The result of quasi-static analysis with nominal and geometrical parameters presents a stiffness of 75.87 N/mm, 25.88% more than the actual one.

The sensitivity analysis showed that the discretization influences the stiffness but even doubling the number of bodies the gap with the actual stiffness cannot be filled up.

On the other hand the influences of thickness and Young’s modulus were investigated and both the parameters could be likely tuned in order to get the desired stiffness (Figure 15 and Figure 16).

The reduction of thickness proved to be the most effective choice. To reach the stiffness of the actual leaf spring a thickness reduction of 7.89% (1 mm) was required, while in case of varying \( E \) a hardly plausible reduction of 22% was necessary. For this reason the first option was preferred for the further dynamic simulations.

![Figure 15](image1.png)

**Figure 15.** Young’s modulus influence on MB model static response.

![Figure 16](image2.png)

**Figure 16.** Thickness influence on MB model static response.
4.2 Simulation of dynamic tests

According to the FEM case, the correlation was firstly carried out with filtered tests results. It proved to be particularly useful to understand the role of mass in simulations. According to the tests, the slope of the characteristic curve changes with the excitation frequency due to inertia forces: since reducing the thickness decreases the mass of the system, the density was increased of 27% to match again the tests results. This phenomenon is visible in Figure 17 and Figure 18, where a damping optimized for the 8 Hz excitation case was used for all the simulations. It is evident that at low frequencies the dissipated energy is strongly underestimated but, similarly to the FEM, case the issue can be solved by using a different coefficient for each frequency.

Table 4 lists the value of the optimum coefficient \( c \) for each simulation. It is noteworthy that the damping coefficient \( c \) has the same decreasing trend with frequency of the FE \( \alpha \) coefficient. The ratio \( \alpha/c \) indeed is constant and equal to 5.

![Figure 17. Simulations with nominal density.](image)

![Figure 18. Simulations with density correction.](image)

<table>
<thead>
<tr>
<th>Frequency [Hz]</th>
<th>c</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.25</td>
<td>0.01</td>
</tr>
<tr>
<td>1</td>
<td>0.0015</td>
</tr>
<tr>
<td>3</td>
<td>0.00075</td>
</tr>
<tr>
<td>5</td>
<td>0.0005</td>
</tr>
<tr>
<td>8</td>
<td>0.0002</td>
</tr>
</tbody>
</table>

Once the density and thus the inertia forces were adjusted the damping coefficient was lowered and the correlation with non-filtered data was carried out. First simulations confirmed the limits associated with using a single damping coefficient for all the blades: with coefficient \( c \) equal to \( 2 \times 10^{-7} \) (approximately the mean value of the first three critical damping coefficient calculated in tests) the component at 120 Hz is preponderant, conversely to what results from experimental tests and FEM simulations.

To get simulations results closer to the experimental response, the damping coefficients of the blades were decoupled. After an optimization process the optimal values listed in Table 5 were found. The values of modal damping used in the model are consistent with the results of the experimental modal analysis as far as the second and the helper leaf is concerned. Modal damping on the main leaf has instead to be significantly increased to match experimental data.

![Table 5. Mean values of modal damping estimated through modal analysis and set in the MB model.](image)

<table>
<thead>
<tr>
<th>Blade</th>
<th>Modal analysis value [%]</th>
<th>Optimized value [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main leaf</td>
<td>0.28</td>
<td>0.45</td>
</tr>
<tr>
<td>Second leaf</td>
<td>0.23</td>
<td>0.2</td>
</tr>
<tr>
<td>Helper leaf</td>
<td>0.57</td>
<td>0.55</td>
</tr>
</tbody>
</table>

The results reported in Figure 19 and Figure 20 are still reveal differences with respect to experimentation: simulations outputs appear over-damped but the main resonances are captured. The FFTs of the force signals are consistent with the experimental ones but lowering the damping coefficient would make arise other components that have no match with tests.
4.3 Computational time

The lack in accuracy of the MBS is balanced by the computational efficiency: two quasi-static cycles are calculated in 7 seconds, while in the matter of dynamic simulations to compute 5 seconds of simulation 50 seconds are required with the first approach, while 10 minutes are necessary with the second one.

5 CONCLUSIONS

In this paper a comparison between two numerical models of a leaf spring suspension has been described and implemented for the scope of determining the best modelling strategy for vehicle dynamics applications. Additionally, an experimental campaign has been carried out and the results have been used to update the model’s parameters.

The first model uses a multi-body approach available in LMS Virtual.Lab Motion where the flexibility of the leaves is modelled by means of a series of discrete elastic elements. The second model is developed in the finite element code Samcef Mecano. It has been shown that this last model matches perfectly the static tests without the need of any model updating. On the other hand the multi-body model needs some tuning to achieve the same accuracy. Both modelling approaches show the need to adapt the damping factor for each test frequency in order to capture correctly the hysteresis loop which indicates that the phenomena is more complex than the simple approach used here and would need a more accurate methodology in order to be able to predict the suspension response to a non-harmonic excitation as in a vehicle dynamics application where the road typically has a profile that will excite at the same time a wide range of frequencies.

Finally, it’s necessary to point out that the multi-body model is easier to build due to the high level of automation present in LMS Virtual.Lab Motion and the possibility of building the model in an assembled and preloaded condition. This is not possible in Samcef Mecano therefore a dedicated analysis case needs to be performed before a static or dynamic analysis. From the computational point of view the multi-body approach outperforms the finite element approach as expected, therefore is more suited for vehicle dynamics applications where the spring is just a component of a larger assembly. Nevertheless the finite element model can be used for more detailed analysis and tuning of the design when a specific target for the stiffness characteristic is required.

ACKNOWLEDGEMENTS

We gratefully acknowledge the European Commission (EC) for their support of the Marie Curie IAPP project 285808 INTERACTIVE (Innovative Concept Modelling Techniques for Multi-Attribute Optimization of Active Vehicles). We would like also to thank Dr. Lorenzo Sanapo for his help with experimental testing.
REFERENCES


Parameter identification of a multibody vehicle model using mathematical optimization

José L. Bueno-López*, María D. Gutiérrez-López*, Javier García de Jalón*

*INSIA
Universidad Politécnica de Madrid
Carretera de Valencia km 7, 28031 Madrid, Spain
joseluis.bueno.lopez@upm.es
md.gutierrez@upm.es

#ETSII and INSIA
Universidad Politécnica de Madrid
José Gutiérrez Abascal 2, 28006 Madrid, Spain
javier.garciadejalon@upm.es

Abstract

Models of real-life mechanical systems often have parameters difficult to measure, calculate or estimate. Even if they can be measured, the process might involve disassembling, which is always costly and laborious. Furthermore, some parameters in the field of vehicle dynamics, such as the torsion stiffness coefficient of the chassis, cannot be calculated without a probably destructive torsion test or with a detailed finite element model of the complete assembly. Therefore, a systematic identification of such parameters using experimental data is highly desirable.

In this paper, the parameters are estimated by the minimization of a mathematical cost function defined as:

$$\min_{\mathbf{b}} \Psi_0 = \sqrt{\int_0^T [(\dot{x} - \hat{x})^2 + (\dot{y} - \hat{y})^2 + (\dot{z} - \hat{z})^2 + (\dot{\phi} - \hat{\phi})^2 + (\dot{\theta} - \hat{\theta})^2 + (\dot{\psi} - \hat{\psi})^2] dt}$$  (1)

which is the root mean square (RMS) value of the difference between the position responses of the model – all translational and rotational bodywork position variables that depend on $t, z, \dot{z}$ and $\mathbf{b}$ – and the experimental results obtained doing a particular maneuver – which are indicated with the hat symbol– and only depend on time $t, z, \dot{z}$ are the position, velocity and acceleration of the independent coordinates of the model and $\mathbf{b}$ is the set of parameters to identify.

The process of the optimization loop is depicted in Figure 1. First of all the objective function and its gradient are evaluated. Therefore, the forward dynamics and the state sensitivities have to be computed. For the former, a semi-recursive method based on a double step Maggi’s formulation is used [1]. The sensitivity analysis is carried out with a hybrid direct-automatic differentiation method which combines the Direct Differentiation Method (DDM) [2] and Automatic Differentiation techniques (AD) [2]. This approach allows computing algorithmically all the terms that would otherwise need to be computed manually. Once solved the forward dynamics and the sensitivity analysis, the evaluation of the objective function and of its gradient is direct.

In order to obtain a realistic solution some optimization constraints are added: grip, wheel contact and box constraints. Grip constraints are based on Pacejka’s model [3] of the side force, $F_s$, and is formulated so that the slope of that force w.r.t the side slip angle, $\alpha$, is always positive. The wheel contact condition formulation is used to enforce positive values of the tire normal forces. Finally, box constraints are imposed on the designed variables in order to prevent unrealistic parameter values.
This method is applied to an IVECO Daily 35C15 industrial van modeled in [4]. Several maneuvers have been performed in order to obtain the necessary experimental data for the parameter identification process: speeding and braking maneuvers with different maximum speeds; circular maneuvers at different speeds; and maneuvers with speed bumps in order to excite the roll movement or the pitch movement.

This method allows identifying the parameters of the model in a rather automatic way by means of using the aforementioned hybrid direct-automatic differentiation method. It also yields to estimate correctly the model’s parameters validating therefore the mathematical model of the vehicle dynamics.

References


Robust High Speed Autonomous Steering of an Off-road Vehicle

Michael Kapp, P. Schalk Els
Faculty of Mechanical and Aeronautical Engineering
University of Pretoria
Hatfield, 0002 Pretoria, South Africa
michkapp@gmail.com schalk.els@up.ac.za

ABSTRACT
This study investigates the use of a Linear Quadratic Self-Tuning Regulator (LQSTR) adaptive control strategy in conjunction with an AutoRegressive with eXogenous input (ARX) model estimator as a steering controller for an off-road vehicle. The LQSTR and the well-known Freund steering controller are implemented in simulation using a validated non-linear multi-body dynamics full vehicle model and tasked with completing an ISO 3888-1 severe double lane change. By considering the path deviation and stability of the controllers, the LQSTR approach is found to be superior. It is able to keep the vehicle stable up to a speed of 115km/h, while the Freund controller loses control of the vehicle at a speed of 90km/h. The adaptive nature of the steering controller and the fact that no vehicle parameters are required during the tuning process proves the LQSTR an elegant steering control solution for use on off-road vehicles that navigate a wide variety of terrains.

Keywords: Autonomous vehicle, Steering controller, Adaptive control, Driver model, Off-road vehicle.

1 INTRODUCTION
According to the National Highway Traffic Safety Administration of the USA, 21.7% of all fatal vehicle collisions occur while negotiating a curve and 15.3% are due to head-on collisions [1]. Driver assistance in the form of lateral collision avoidance systems may be able to reduce the number of fatalities in these instances by reacting quicker than an average driver is able to. However, the wide range of operating conditions an off-road ground vehicle has to contend with provide a significant challenge to control engineers, especially when the vehicle is required to be stable at high speeds while travelling on unpredictable surfaces. A series of algorithms have been proposed by researchers to safely steer a vehicle through a dynamic manoeuvre, ranging from conventional techniques to intelligent systems theory; some have been implemented with varying levels of success.

An elegant solution to this non-linear problem may be the use of adaptive control. In adaptive control, the dynamics of the process is estimated on-line using either a reference model or a black-box approach. This allows the controller to be adjusted to deliver the optimal performance under the current operating conditions [2]. Although the use of adaptive control in vehicle steering is not a new concept, very little literature is available on the topic compared to other approaches. Some examples include adaptive strategies implemented to reject the disturbance caused by wind gusts on a vehicle [3], as well as in the form of a lateral controller capable of path following up to 25 m.s^{-2} [4].

This paper investigates the use of adaptive control in vehicle applications by considering a Linear Quadratic Self-Tuning Regulator (LQSTR) for use as a path following steering controller for an off-road vehicle. By implementing this black-box approach, no prior vehicle parameters are required other than the tuning parameters of the Linear Quadratic Regulator (LQR) performance function. This should allow a greater variety of stable operating conditions and
could make the steering controller more robust against changes in vehicle parameters and other external disturbances.

2 CONTROLLER AND LQSTR DESIGN

An LQSTR [2] is an adaptive strategy that utilises the stability of the conventional LQR approach and combines it with a model estimator to yield a controller capable of adapting its tuning parameters to changes in the controlled plant. In this study, it is used to control the yaw rate of the vehicle.

2.1 Controller overview

Vehicle yaw control and lateral position control are the fundamental components of a path following controller. This study prioritises the first, as control of the yaw angle ($\psi_d$) and especially the yaw rate ($\dot{\psi}_d$) of the vehicle is directly related to the stability of the vehicle and the safety of its occupants. Given the known first order response between the steering angle of the front wheels and the yaw rate of the vehicle when operating in the linear region, this study uses the front wheel steering angle as the manipulated variable and the yaw rate of the vehicle as the controller variable. This is also done to ease the measurement process and reduce the amount of signal conditioning required. Conversion of the desired yaw angle ($\psi_d$), determined by the curvature of the desired path ($\kappa_d$), to a desired yaw rate is however required and is performed under the assumption of a constant acceleration over the sampling period (Equation 1).

$$\Delta \psi = \frac{d\psi}{dt} \Delta t + \frac{1}{2} \frac{d^2\psi}{dt^2} \Delta t^2$$

(1)

Figure 1 illustrates the parameters used in this study, while Figure 2 shows the general structure of the controller.

Figure 1: Definition of vehicle parameters

The nature of the vehicle system requires the desired yaw rate parameter ($\dot{\psi}_d$) to be calculated at a preview point ahead of the vehicle ($d_{path}$). This point is calculated using a set preview time ($\tau_{path}$) and the forward velocity of the vehicle ($\dot{x}$). Similarly the forecasted lateral error $E_{lat}$ is calculated at $d_{lat}$. Small deviations from the desired path ($E_{lat}$) are considered as a secondary objective and is integrated in the yaw angle reference through the use of a simple gain. Although not optimal across all vehicle speeds, this is considered a compromise to prevent conflicting objectives between the yaw angle controller and the lateral position controller. A
gain of 1 was found suitable for this study, but will require adjustment according to the vehicle used.

The LQSTR used in this study consists of two main components: (1) the AutoRegressive model with eXogenous input (ARX) estimator and (2) the LQR. The estimator fits an ARX model to one second of measured steering angle and yaw response data. These calculated ARX parameters are then used to update the LQR gains at a frequency of 5Hz and to determine the required steering angle of the front wheels at a control frequency of 20Hz [5]. A fixed window approach is used to limit the effect of inaccurate model estimations on the performance of the steering controller. These parameters were chosen as the lowest frequencies at which sufficient controller performance was observed and may be adapted to the application. Figure 3 illustrates the LQSTR and its integration with the vehicle system.

2.2 ARX model estimator

Regression theory, especially ARX [2, 6], is commonly used in the process of system identification [7]. An ARX model describes the output $\hat{y}_k$ of a system as a linear function of previous outputs $y_k$ (the autoregressive portion) and previous and current inputs $u_k$ (the exogenous input portion). Equation 2 illustrates this mathematically, where $p$ is the number of autoregressive terms, $b$ the number of exogenous inputs and $\varphi_i$ and $\eta_i$ the autoregressive and input weights respectively. A residual error term $\epsilon_t$ is used to account for imperfect fits.

$$\hat{y}_k = \sum_{i=1}^{p} \varphi_i y_{k-i} + \sum_{i=0}^{b} \eta_i u_{k-i} + \epsilon_t$$  \hspace{1cm} (2)

The ARX model can also be described as an Infinite Impulse Response (IIR) filter and can similarly be written as a linear, time-invariant transfer function. By utilising this ability and choosing the model orders correctly, the yaw rate response of the vehicle as a result of a steering input can thus be estimated by fitting Equation 2 to measured input-output data.

A first order transfer function in discrete time is shown in Equation 3. Although it is known that the yaw rate response should be first order when operating in the linear region, the system is
expected to function in the non-linear region as well. For this reason, an additional discrete pole is added to increase the system’s dependence in the autoregressive portion (dominated by the momentum of the vehicle). Similarly, as the steering effect lessens in the non-linear region, the input dependency of the system is decreased by removing a discrete pole from the transfer function. This yields the transfer function shown in Equation 4.

\[
H(z) = \frac{\eta_0 z^{-n} + \eta_1 z^{-1-n}}{1 + \varphi_0 z^{-1}} \tag{3}
\]

\[
H(z) = \frac{\eta_0 z^{-n}}{1 + \varphi_0 z^{-1} + \varphi_1 z^{-2}} \tag{4}
\]

It has also been proven that the majority of autoregressive processes can be represented by a second order (AR(2)) model [2]. This study uses a linear least squares approach to fit Equation 2 to the input-output data using an auto-regressive order of 2 and an input order of 1 [5].

2.3 Linear Quadratic Regulator

The LQR is based on an optimal pole placement process and conventionally requires full state feedback, either directly or from an observer. However, by writing the ARX model in the discrete controller canonical state space form with previous outputs as states, direct full state feedback is achieved without the need for a state observer. This is illustrated in Equations 5-9.

\[
X_k = AX_{k-1} + Bu_k \tag{5}
\]

\[
Y_k = CX_k \tag{6}
\]

\[
A = \begin{bmatrix}
-\varphi_0 & -\varphi_1 \\
1 & 0
\end{bmatrix} \tag{7}
\]

\[
B = \begin{bmatrix}
1 \\
0
\end{bmatrix} \tag{8}
\]

\[
C = \begin{bmatrix}
\eta_0 \\
0
\end{bmatrix} \tag{9}
\]

Since the LQR is an optimal controller, it requires a performance function during the design process to determine the characteristics or sensitivity of the controller. This study uses a performance function \( J_{LQR} \) to maintain a balance between setpoint \( \dot{\psi}_{sp} \) following through the \( Q_{LQR} \) parameter and steering angle \( \delta \) utilisation through the \( R_{LQR} \) parameter. These parameters are tuned by performing manoeuvres in simulation and are vehicle dependent. A combination of \( Q_{LQR} = \begin{bmatrix} 9 & 0 \\ 0 & 1 \end{bmatrix} \) and \( R_{LQR} = 1 \) was found to be optimal for the current vehicle configuration. Equation 10 shows the performance function.

\[
J_{LQR} = \sum_{n=0}^{\infty} [(\dot{\psi}_{sp} - \psi_a)^T Q_{LQR} (\dot{\psi}_{sp} - \psi_a) + \delta^T R_{LQR} \delta] \tag{10}
\]

Once both the model and performance function is known, the dynamic Riccati equation (Equation 11) is solved by iterating in time until the solution converges.

\[
P_{k+1} = Q_{LQR} + A^T (P_k - P_k B (R_{LQR} + B^T P_k B)^{-1} B^T P_k) A \tag{11}
\]

Upon conversion, the LQR gain can be calculated as shown in Equation 12.
The LQSTR is therefore comparable to an intelligent gain scheduling controller where, instead of using a predefined look-up table, the controller gain is calculated based on the current dynamics of the plant. The final control law for implementation is illustrated in Equation 13.

$$ K_{LQR} = R_{LQR}^{-1}B'P $$

At each control interval (0.05s), the control law is applied to determine the steering angle of the front wheels. The control gain matrix is also updated at the model update frequency (5Hz) by recalculating the Riccati equation using the updated ARX model estimate to yield an adaptive controller.

3 SIMULATION STUDY

3.1 Vehicle simulation model

The simulations are performed using a 16 degree of freedom non-linear multi-body dynamics model of a Land Rover Defender 110 Tdi developed in MSC ADAMS View. The full vehicle model incorporates experimentally determined body torsions, moments of inertia and suspension characteristics, along with joints and suspension geometry as they appear on the actual vehicle. An experimentally determined Pacejka '89 tyre model [8] is used to characterise the lateral force generation capabilities of the tyres [9], while diabatic process theory is used to include the temperature dependency of the suspension system. The model has been fully validated for lateral, longitudinal and vertical dynamics [10, 11].

Els fitted the vehicle with a hydro-pneumatic suspension system, also known as the $4S_4$, in an effort to eliminate the ride-handling compromise present in suspension design [12]. This system allows the driver to switch between a comfort mode (low spring stiffness and damping) and a handling mode (high spring stiffness and damping), which represent the two extremes of suspension system configurations. These features are also included in the full vehicle model and are used to alter the handling dynamics of the vehicle in an effort to determine the controllers’ robustness against changes in the vehicle parameters.

The steering angle of the vehicle is limited to 30° to each side as per the geometry of the actual vehicle, while the front wheel steering rate of the vehicle is limited to 15°/s. This corresponds to a steering wheel rate of 720°/s, the maximum that a skilled driver can achieve while performing a fish hook test in the vehicle.

3.2 Simulation setup

The performance of the driver models is evaluated by navigating an ISO 3888-1 severe double lane change (DLC) [13] in simulation (desired path provided in Figure 4). This is done using a co-simulation setup between MSC ADAMS View (responsible for solving all dynamics except suspension forces) and Mathworks Simulink (responsible for all other calculations, including driver model and suspension forces). The model is solved at a frequency of 200Hz.
Evaluation is performed by considering the root-mean-squared error (RMSE) of the vehicle center of mass from the desired path, as well as the maximum stable path following speed achieved by the controller. The RMSE is calculated as shown in Equation 14.

\[
RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left( \min \left( \sqrt{\left( y_{path_i} - c_{y_i} \right)^2 + \left( x_{path_i} - c_{x_i} \right)^2 } \right) \right)^2}
\]  

(14)

### 3.3 The Freund steering controller

The Freund steering controller was developed by Freund and Mayr [14] and is a well-known algorithm commercially used in software such as veDYNA [15]. It consists of a non-linear vehicle model, tyre model and a steering saturation function. The saturation function prevents the controller from steering the vehicle too vigorously at higher speeds and creating an unstable situation. Although it is known for general stability, the Freund controller requires the tyre-force response of the vehicle to be known a-priori. For the purpose of this study, it was tuned to provide the same path following accuracy as the LQSTR approach at a vehicle speed of 60km/h. This allows for comparable results between the two controllers.

### 3.4 Results and discussion

The RMSE values measured during the simulation study are plotted against vehicle speed for each controller and suspension configuration in Figure 5. Both controllers are found to be robust against changes in the handling characteristics of the vehicle, with differences only being observed close to the limits of each individual controller. The steering controllers are also found to deliver similar performance at vehicle speeds below 60km/h where the vehicle dynamics are fairly linear. However, when the vehicle speed is increased above 60km/h, a significant difference in performance is observed.

The performance of the Freund controller deteriorates immediately above the initial tuning speed of 60km/h and is able to maintain vehicle stability up to a vehicle speed of 90km/h. A maximum RMSE of 0.9m is measured here (comfort mode suspension). However, the RMSE of the LQSTR remains fairly constant up to a vehicle speed of 100km/h, where an RMSE of 0.25m is recorded. This illustrates the adaptive approach’s ability to adapt to changes in the lateral response of the vehicle as it enters the non-linear operating regime. At higher speeds, the value deteriorates up to a speed of 115km/h (RMSE of 1.3m), after which the vehicle becomes unstable. This is also known to be close to the maximum speed the specific vehicle is able to complete the DLC manoeuvre.
4 CONCLUSION

The use of an adaptive control approach known as an LQSTR as a steering controller for an off-road vehicle was investigated and its performance compared to that of a well-known driver model. Results show the LQSTR to be superior to the Freund controller in steering an off-road vehicle through a DLC manoeuvre at different speeds using the same set of tuning parameters. Although the same performance is achieved at a speed of 60km/h, the Freund controller quickly loses accuracy at higher speeds and is unable to maintain vehicle stability through the manoeuvre at speeds above 90km/h. The LQSTR is however able to accurately follow the required path up to 100km/h and maintain vehicle stability up to a speed of 115km/h (the handling limit of the vehicle). There is thus merit in using adaptive control approaches in the lateral control of an off-road vehicle, as it provides a robust response without the need for any prior knowledge about the vehicles operating parameters or driving environment and thereby eliminating the need for complex observers. This will be especially advantageous in off-road vehicles that travel on a variety of surfaces that cannot always be captured using mathematical or empirical models.

REFERENCES


Simulation of friction-induced vibrations using elastic multibody models

Igor Iroz, Michael Hanss, Peter Eberhard

Institute of Engineering and Computational Mechanics
University of Stuttgart
Pfaffenwaldring 9, 70569 Stuttgart, Germany
[igor.iroz, michael.hanss, peter.eberhard]@itm.uni-stuttgart.de

ABSTRACT
In this paper, by the use of elastic multibody dynamics and a master-slave contact approach with penalty formulation, computationally efficient time integrations of a brake system are performed for constant and time-dependent input parameters. As a result, the amplitudes of the friction-induced vibrations and the contact forces at the disc-pad interfaces are predicted. Besides, system outputs are viewed in phase diagrams and the creation of a stable limit cycle for a low friction coefficient is identified. In this way, conclusions on the stability of the system are drawn and statements based on frequency-domain analyses are complemented. Finally, a distinct need for a new criterion that quantifies the squeal propensity of such systems in the time domain is identified.

Keywords: Elastic multibody system, self-excited vibration, friction-induced vibration, brake squeal, penalty formulation.

1 INTRODUCTION
Simulation of friction-induced vibrations still represents a challenging topic for engineers. These self-excited oscillations are present in many day-to-day applications and they can yield, among others, uncomfortable sound emission or unpredictable mechanical excitations. Examples of such phenomena are found in the squealing of brakes, the chattering of a cylinder being manufactured for a spindle, or the scraping of chalk on a blackboard. Since the 1960s, the problem of disc-brake squeal has arisen as particularly challenging for the automotive industry [1].

Commonly, in order to detect the instabilities, investigations in the frequency domain are performed. A very well-known procedure is the method of complex eigenvalue analysis (CEA) [2]. This method is based on the linearization of the nonlinear equations of motion around a working point, which is afterwards classified as stable or unstable depending on the real part of the provided complex eigenvalues. However, the method tends to overestimate the number of unstable frequencies found in experimental investigations and does not predict the displacements since the delivered complex eigenmodes are normalized.

For the purpose of calculating the magnitude of the deformations, time domain investigations are a necessary but expensive solution. In a first attempt, integrating large finite-element (FE) models over time, such as the one in Figure 1, is neither promising nor affordable because of long calculation times [2, 3]. At this point, using elastic multibody systems (EMBS) based on the floating frame of reference approach [4, 5] is advantageous because it enables the efficient coupling of large nonlinearities resulting from rotation and contact with small, linearly described deformations. Moreover, this approach fits well to the usual process chain in mechanical design and benefits from the widespread and detailed FE modeling of components.

Contact modeling between interacting bodies plays a decisive role, too. A master-slave approach with a penalty formulation is applied by coupling the contact module proposed in [6] with the EMBS program Neweul-M². As a result, penetration of slave nodes into master faces and corresponding normal contact forces are calculated. In addition, the interface is extended to include slave and master nodal velocities, tangential directions and different friction laws which allow
adding tangential contact forces to the system. In between, a rough detection algorithm manages
an efficient handling of slave and master nodes and triggers signals that interrupt time integration.
In this way, contact pairs or even multibody systems itself are updated and new time investigations
with new contact situations are restarted.

2 CONTACT SIMULATION BASED ON ELASTIC MULTIBODY SYSTEMS

Multibody systems (MBS) are well suited to simulate the kinematics and dynamics of mechanical
systems composed of a number of rigid bodies that undergo large, nonlinearly described mo-
tions [4]. In many engineering tasks, e.g. in vibration analyses, elastic deformations have to be
considered, and thus, the basic theory of MBS is extended to include small deformations in elastic
bodies. This results in elastic multibody systems, which are particularly efficient for dynamic time
simulations.

2.1 Floating frame of reference approach and kinematics

Regarding the modeling of elastic multibody systems, the floating frame of reference approach is
used [5]. This approach is well suited for the study of oscillations in general and friction-induced
vibrations in particular, since it encourages an efficient decoupling of large nonlinearly described
motions and small linearly described deformations.

In this approach, a so-called floating frame of reference is attached to the elastic body in order to
decouple rigid body motion and elastic deformation. On the one hand, the translation vector \( \mathbf{r}_{\text{IR}} \)
and the rotation matrix \( \mathbf{S}_{\text{IR}} \) of the reference system \( \mathbf{K}_R \equiv \{ \mathbf{r}_{\text{IR}}, \mathbf{S}_{\text{IR}} \} \) represent the large, nonlinearly
described rigid motion of the body with respect to the inertial system \( \mathbf{K}_I \). On the other hand, the
small, linearly described elastic deformation at an arbitrary point P is described in the reference
system by considering two configurations, namely, the reference configuration and the current or
deformed configuration.

In the following, the reference system is used for describing the kinematic magnitudes of an
EMBS. As shown in Figure 2, the relative position of an arbitrary point P in the reference config-
uration is described by the vector \( \mathbf{R}_{\text{RP}} \). In an equivalent way, after being subjected to an elastic
deformation, \( \mathbf{r}_{\text{RP}} \) represents the relative position of P in the current configuration. Besides, the
deformation in between these two configurations is described by the vector \( \mathbf{u}_P \), which leads to the
basic relation

\[
\mathbf{r}_{\text{RP}} = \mathbf{R}_{\text{RP}} + \mathbf{u}_P(\mathbf{R}_{\text{RP}}, t).
\]
By the use of global Ritz functions, an advantageous separation of the dependent variables is achieved [5]. The deformation of the elastic body is then given as

\[ r_{RP} = R_{RP} + \Phi_P(R_{RP}) \cdot q(t), \]  

(2)

where \( \Phi_P(R_{RP}) \) are the discretization-dependent elastic modes and \( q(t) \) are the time-dependent elastic coordinates of the EMBS simulation. Now, the absolute position of the point P is characterized in the current configuration by

\[ r_{IP} = r_{IR} + r_{RP} = r_{IR} + (R_{RP} + \Phi_P(R_{RP}) \cdot q(t)), \]  

(3)

and after differentiating Equation (3) with respect to time, the absolute velocity reads as

\[ v_{IP} = v_{IR} + \dot{\omega}_{IR} \cdot r_{RP} + \dot{r}_{RP} = v_{IR} + \dot{\omega}_{IR} \cdot R_{RP} + \Phi_P(R_{RP}) \cdot \dot{q}(t). \]  

(4)

As a result, a complete kinematic description of an elastic body is available. Amongst others, Equations (3) and (4) are used when quantifying the vibration amplitudes at selected points or when calculating the relative velocities at the contact interface, as performed in the next subsection for a friction-affected system.

### 2.2 Contact modeling and contact force computation

The modeling of contact interfaces plays a decisive role in the dynamic simulation of friction-induced vibrations. In this paper, a master-slave approach with a linear penalty function [6] is used in order to calculate the penetration and the contact forces influencing the dynamics of the EMBS.

Based on the kinematic discretization described in Subsection 2.1, absolute positions \( r_{IP}^1 \) and \( r_{IP}^2 \) and velocities \( v_{IP}^1 \) and \( v_{IP}^2 \) for two node sets \( c_1 \) and \( c_2 \) at the contact interface are calculated in the inertial frame. In the master-slave procedure, the penetration of a slave node \( s \in c_1 \) into a master surface \( m \) composed by three or four master nodes \( m_i \in c_2 \) is checked for all master-slave combinations, as shown in Figure 3 for a single quadrangular contact pair. If a two-pass algorithm is used, the master-slave definition is inverted, i.e. \( s \in c_2 \) and \( m_i \in c_1 \), and a weighting factor \( w \in [0, 1] \) is defined [6].

For the corresponding slave-master contact pair, if the slave node \( s \) penetrates into the master surface \( m \), the penetration gap \( g \) and velocity \( \dot{g} \) in the normal direction \( n \) and local relative velocity
Figure 3: Quadrangular contact pair in the master-slave approach.

\( \mathbf{v}_{\text{rel}} \) in the tangential direction \( \mathbf{t} \) are computed. Normal and tangential direction force laws define the behavior of the contact interface at this point.

For the normal direction, a penalty formulation is used, which can be defined as a spring-damper force element of the form

\[
f_n = (c_p |g| + d_p |\dot{g}|) \mathbf{n}
\]  

acting in the normal direction \( \mathbf{n} \) with the stiffness and damping parameters \( c_p \) and \( d_p \), respectively. The definition of the penalty factors is not trivial and their variation alters significantly the results. In this article, for example, a linear, gap-dependent force law of the form \( f_n = c_p |g| \) is proposed [6], but in order to capture the effect of orthotropic contact materials, like for brake pads, a third order polynomial has also been used in the literature [7].

In order to reproduce self-excited vibrations, for the tangential direction a classical Coulomb friction of the form

\[
f_t = \mu |f_n| \mathbf{t}
\]  

acting in the tangential direction \( \mathbf{t} \) is modeled, where \( \mu \) is a friction coefficient which is constant or dependent on relative velocity \( \mathbf{v}_{\text{rel}} \). Particularly problematic is the singularity around \( \mathbf{v}_{\text{rel}} = 0 \), which yields undesired local instabilities when stick phenomena occurs in the contact interface. For that purpose, regularized Coulomb or Stribeck curves that allow a smoother transition between static and dynamic friction can be used [8]. Tribological effects also play an important role, and thus, can be represented in the tangential plane and in the normal direction [9]. Further detailed fundamentals on contact mechanics are found in the literature, where a number of analytical force laws are deduced and investigated [10].

Based on the normal and tangential contact laws, the resulting nodal contact force \( \mathbf{f}^s \) for each slave node \( s \) is calculated. In order to satisfy the static equilibrium in the contact pair, the nodal force is distributed into the corresponding master nodes \( m_i \) by using appropriate local ansatz functions, resulting in the nodal contact forces \( \mathbf{f}^{m_i} \). This procedure is repeated for all the master-slave pairs at the interface defined by the node sets \( c_1 \) and \( c_2 \) and results in the contact-force vectors \( \mathbf{f}^{c_1} \) and \( \mathbf{f}^{c_2} \) for the bodies in contact. Finally, nodal contact forces are transformed into the modal space using the shape functions \( \Phi \), see Equation (2), and are added to the right-hand side of the equations of motion of the system described in the next subsection.

### 2.3 Equations of motion of an elastic multibody system

Based on the fundamentals of multibody systems, the Newton and Euler equations are used to describe the dynamics of a rigid body [4, 5]. By the use of the principles of mechanics, such as the
principle of virtual work, the principle of d’Alembert or the principle of Jourdain, the equation of motion for one elastic body is deduced and reads as
\[
M \ddot{z}_{II} = \begin{bmatrix}
m & \text{sym.} \\
m \dot{c} & J \\
C_t & C_r & M_e
\end{bmatrix} \begin{bmatrix}
\dot{v}_{\text{IR}} \\
\dot{\omega}_{\text{IR}} \\
\ddot{q}_e
\end{bmatrix} = \mathbf{h}_o + \mathbf{h}_e + \mathbf{h}_a,
\] (7)

where \(v_{\text{IR}}\) and \(\omega_{\text{IR}}\) are the translational and rotational velocities of the floating frame of reference, \(q_e\) are the reduced elastic coordinates in the modal space, \(h_o\) are the generalized inertial forces, \(h_e\) are the internal forces and \(h_a\) are the external forces. Contact forces described in Subsection 2.2 are added to the the right-hand side of Equation (7) into the vector of external forces \(h_a\) after being transformed into the modal space.

The generalized mass matrix \(M\) possesses a \(3 \times 3\)-block structure. The first and second block rows of the mass matrix \(M\) represent the translational and rotational components of the rigid coordinates, while the third block row is related to the elastic coordinates. Thereby, the mass \(m\) and the moment of inertia \(J\) are coupled by the center of gravity of the body \(c\) and form a block that is also coupled by \(C_t\) and \(C_r\) to the reduced elastic mass matrix \(M_e\). The internal forces \(h_e\) represent an important part of the linear elastic description of the EMBS and read as
\[
h_e = \begin{bmatrix}
0 \\
0 \\
h_{ee}
\end{bmatrix} = \begin{bmatrix}
0 \\
0 \\
D_e \cdot \dot{q}_e + K_e \cdot q_e
\end{bmatrix},
\] (8)

where \(D_e\) and \(K_e\) are the reduced damping and stiffness matrices, respectively. In this way, a full description of the nonlinear equations of motion is available for a friction-affected EMBS.

3 APPLICATION TO A SYSTEM AFFECTED BY FRICTION-INDUCED VIBRATIONS

In this paper, the disc brake shown in Figure 4 is investigated [11]. The system consists of a brake disc, two brake pads and a caliper which are modeled as finite-element structures. Regarding the material, the disc and the caliper are made of steel, while the brake pads are modeled as an orthotropic composite material with structural damping and different properties in the out-of-plane direction. In Table 1, the most important material characteristics of the FE structures are summarized.

<table>
<thead>
<tr>
<th>structure</th>
<th>material</th>
<th>type</th>
<th>modulus (E)</th>
<th>Poisson (\nu)</th>
<th>density (\rho)</th>
<th>damping (\varepsilon)</th>
</tr>
</thead>
<tbody>
<tr>
<td>disc</td>
<td>light steel</td>
<td>isotropic</td>
<td>42 GPa</td>
<td>0.3</td>
<td>4800 kg/m³</td>
<td></td>
</tr>
<tr>
<td>caliper</td>
<td>steel</td>
<td>isotropic</td>
<td>210 GPa</td>
<td>0.3</td>
<td>7800 kg/m³</td>
<td></td>
</tr>
<tr>
<td>pads</td>
<td>composite</td>
<td>orthotropic</td>
<td>0.5 GPa</td>
<td>0.4</td>
<td>2500 kg/m³</td>
<td>0.08%</td>
</tr>
</tbody>
</table>

In the presented friction-affected system, the disc rotates with constant rotational velocity \(\omega\) by means of a rotational joint which is aligned to the \(y\)-axis of the inertial system. The pads are pressed against the disc by means of 25 separate nodal forces \(F_n\) that create an equivalent distributed pressure at each surface. Lower and upper pads exhibit displacements in \(y\)-direction, and their rigid body motion is described by the coordinates \(y_{\text{lpad}}\) and \(y_{\text{upad}}\). Furthermore, each pad is connected to the caliper via 25 springs with stiffness \(k = 200\) kN/m and act in the \(y\)-direction in order to couple pad and caliper dynamics. Regarding the caliper, it is connected to the inertial system on its lower-pad side by four springs \(k_c = 1\) kN/m acting also in the \(y\)-direction. In this case, out-of-plane motion is represented by the generalized coordinate \(y_{\text{cali}}\), making the total number of rigid body coordinates equal to three.
The previously described rotational velocity $\omega$ and nodal forces $F_n$ represent the parameters of the brake system, and thus, are considered as system inputs. In this investigation two cases are studied regarding the time dependency of the parameters. First, constant inputs $\omega = 1 \text{ rad/s}$ and $F_n = 16 \text{ N}$ are considered. Second, time-dependent signals for both parameters are defined, as shown in Figure 5. In this case, the parameters start from zero and increase linearly at different points in time until they reach the values from the first case. This description enables a smooth transition from the initial conditions into the stationary state [7, 12].

As output signals, the out-of-plane motions of four nodes at the caliper surface are selected. This selection is motivated by the fact that these points are accessible by vibration-measurement instruments like laser Doppler vibrometers (LDV), which could be used to validate simulative results in future investigations. By combining output positions and velocities into state-space phase diagrams, the creation of stable limit cycles used to predict instabilities is observed.
3.1 Implementation details and software tools

In order to study squealing brake systems, it is a common practice to perform complex eigenvalue analyses based on finite-element models. Time-domain investigations based on elastic multibody systems benefit from the detailed geometry definition of the frequency-domain analyses. In a first pre-processing step, elastic bodies are produced by importing the FE structures from the CEA and then reducing the number of elastic degrees of freedom with the program MatMoremb [13]. Regarding the reduction, classical modal reduction is applied and the deformation of each elastic body is described as a linear combination of 20 elastic coordinates and their corresponding normal eigenmodes. The eigenfrequencies corresponding to the eigenmodes cover the frequency spectrum from 0 kHz to 6 kHz, which is the range of interest attributed to high-frequency squeal in disc-brake systems [1].

At the reduction step, however, more advanced methods that lead to an efficient reduction of the elastic coordinates can be used. For example, projections onto Krylov subspaces, which are calculated based on the interpolation of the transfer function of a system by using so-called moment-matching, improve the reduced transfer function in a selected frequency range [14]. Moreover, model order reduction methods based on singular value decomposition (SVD) and Gramian matrices are well suited if certain deformation patterns have to be considered [15]. Modern methods can also be combined with other classical, well-established methods, such as the component mode synthesis (CMS) method [16], if displacements in body-coupling directions are to be included by the use of constraint modes.

Once reduced, elastic bodies are exported to different formats according to the EMBS program to be used. In this paper, elastic bodies are exported in the form of standard input data (SID) files and the symbolic EMBS-program Neweul-M2 [17] is used. The program enables a detailed definition of multibody systems with rigid and elastic bodies and includes features like position and velocity constraints or contact definition. Furthermore, a number of formulations and solvers that suit different types of MBS are available [18]. Once the system is defined, the nonlinear or linear equations of motions of the system are calculated and exported to external files for an efficient system evaluation.

4 SIMULATION RESULTS

In this section, simulation results of the elastic multibody system presented in Section 3 are summarized. In order to investigate the system in the time domain, the nonlinear equations of motion are integrated from \( t_0 = 0 \) ms to \( t_{\text{end}} = 100 \) ms using an explicit, fourth-order Runge-Kutta method with a fixed time step of \( \Delta t = 10^{-6} \) s. This yields a total number of 100,000 evaluations. Initial positions and velocities for both rigid and elastic coordinates are set to zero and both constant and time-dependent input signals \( i(t) \) for the parameters \( \omega \) and \( F_n \) are considered.

As far as contact definition is concerned, two disc-pad interfaces are defined. For each pad 25 contact nodes resulting in 16 contact faces are selected, while for each disc contact interface 48 nodes and 35 faces are used. Since a weighting factor of \( w = 0.5 \) is used, two passes are performed for each contact pair. For the normal direction contact, a linear penalty function is used with a factor of \( c_p = 10^8 \) N/m, while in the tangential direction, Coulomb friction is applied with a constant friction coefficient of \( \mu = 0.1 \).

The evolution of the generalized coordinates for the two cases \( i = \text{const.} \) and \( i(t) \) is shown in Figure 6. On the one hand, out-of-plane motions of the brake pads \( y_{\text{pad}} \) and \( y_{\text{upad}} \) are very similar due to the symmetry of the problem. On the other hand, \( y_{\text{cali}} \) shows a smoother, lower-frequency motion than the pads, probably because the high-frequency dynamics of the contact are filtered by the springs at the caliper-pad interface. From \( t = 5 \) ms on, a steady state is reached and \( y_{\text{pad}} \) and \( y_{\text{upad}} \) slightly oscillate around \( \pm 50 \) \( \mu \)m, whereas for \( y_{\text{cali}} \) a lower frequency and a \( \pm 5 \) \( \mu \)m oscillation are observed.
The light lines correspond to the case $i(t)$ and, as expected, much slower dynamical behavior is observed. In the interval $t \in [0, 10]$ ms, linear rotation $\omega(t)$ and no nodal forces $F_n(t)$ are applied, and thus, no motion is detected for the coordinates $y_{lpad}$, $y_{upad}$ and $y_{disc}$. From $t = 15$ ms on, nodal forces are progressively applied and the rigid coordinates are allowed to reach the steady-state positions by $\pm 50 \mu m$. Regarding the caliper motion, no significant oscillation is detected due to the smoother application of the inputs.

In a post-processing step, contact forces are calculated and the rigid, out-of-plane components are represented for the three bodies in contact, see Figure 7. In the stationary state, for the constant case $i = \text{const.}$, the contact forces in the pads $f_{lpad}$ and $f_{upad}$ oscillate around $\pm 135$ N with a maximum value of $f_{max} = \pm 175$ N and a minimum value of $f_{min} = \pm 95$ N. Furthermore, in the transient phase, peaks of $2.8$ kN and some contact intermittencies are observed because the pads lose contact. Regarding the force on the disc $f_{disc}$, an amplitude of $30$ N is predicted and a lower frequency is observed in comparison to the pads.

Figure 6: Generalized rigid body coordinates for constant and time-dependent inputs (light lines).

Figure 7: Contact forces in $y$-direction for constant and time-dependent inputs (light lines).
For the light lines in case $i(t)$, bodies first enter in contact at $t = 21$ ms once the clearance between disc and pads disappears. Because of the progressive application of the nodal forces $F_n(t)$, the transient phase results look softer and maximal peaks of just 250 N are observed in the pads. Contact is barely lost between the contact interfaces and, in the steady state, the contact forces $f_{pad}$ and $f_{upad}$ stabilize, again, around $\pm 135$ N. The contact force at the disc $f_{disc}$ exhibits a similar amplitude for $t > 30$ ms but a higher frequency and, if thoroughly observed, a short period $26$ ms $< t < 30$ ms with smaller amplitudes.

The measured outputs for a caliper node are plotted in the phase diagram shown in Figure 8. For a friction coefficient value of $\mu = 0.1$, stable limit cycles are identified, where the size of both limit cycles correspond with the amplitudes of $y_{cali}$ observed in Figure 6 and with the inputs of the case $i = \text{const.}$ and $i(t)$. These phenomena may indicate that the corresponding frequencies play a decisive role in the squeal generation of the system. For this reason, a fast Fourier transform (FFT) over the velocity of the output is performed and some predominant peaks at 1.68 kHz, 2.34 kHz, 3.96 kHz and 4.69 kHz are detected for both cases, see Figure 9.

Finally, the extracted peaks are compared with the results of the CEA. For the described system,
three unstable complex conjugate pairs of eigenfrequencies are found in the frequency intervals $f_1 \in [782, 791]$ Hz, $f_2 \in [1.60, 1.61]$ kHz and $f_3 \in [2.76, 2.78]$ kHz [11], being the second interval the one that most approximates to the first peak. As a result, the EMBS simulation confirms the instability found in the CEA for the second complex eigenvalue and discards the other two eigenvalues as possible sources for instabilities.

5 CONCLUSIONS

In this paper, elastic multibody dynamics is introduced to the simulation of friction-induced vibrations. The advantageous decoupling of large, nonlinearly described motions and small, linearly described deformations enables an efficient contact modeling and the computation of the contact forces and displacements at the interfaces where the self-excited vibrations are generated. It is shown that time-domain investigations represent an important tool for the prediction of vibration amplitudes, and parallel to well-established frequency-domain analyses, a new method and a process chain are presented. Finally, the method is applied to a simplified disc brake, and together with the delivered time-domain results, a new way for confirming or discarding instable frequencies found in complex eigenvalue analyses is proposed.

In future research, industrially relevant brake systems will be investigated by means of elastic multibodies. Further efforts will focus on the definition of robust criteria that help to determine and classify the squeal propensity of brakes in the time domain. The understanding of the underlying phenomena in friction-induced vibrations represents, as well, a challenging question that still needs to be answered.

REFERENCES


Use of flexible models in extended Kalman filtering for vehicle body force estimation

Sebastiaan v. Aalst#, Frank Naets#, Johan Theunissen*, Wim Desmet#

# Department of Mechanical Engineering
KU Leuven - University of Leuven
Celestijnenlaan 300, 3000 Leuven, Belgium
sebastiaan.vanaalst@kuleuven.be
frank.naets@kuleuven.be
wim.desmet@kuleuven.be

* Flanders Make
Strategic Research Centre Manufacturing Industry
Oude Diestersebaan 133, 3920 Lommel, Belgium
sebastiaan.vanaalst@flandersmake.be
jtheunissen@tenneco.com

ABSTRACT
The forces acting on a vehicle are an important area in vehicle dynamics research. A direct measurement of these forces, however, is not feasible for mass-produced vehicles. This has motivated the use of model-based estimation techniques such as the Kalman filter. In this work the discrete-time augmented extended Kalman filter is employed to perform a combined estimation of the current vehicle state and the input forces, such as the tyre forces. Currently, the general approach is to use simplified ad-hoc models for the Kalman filter. In vehicle dynamics applications this can lead to an unobservable system which makes it impossible to estimate the individual tyre forces. In order to solve this issue, this work proposes to take the flexibility of the vehicle body into account in the model for the Kalman filter. This adds an effect to the model that allows to differentiate between the contributions of all the different input forces. The model is derived according to the floating frame of reference formulation and a reduced order finite element model is used to describe the flexibility of the vehicle body. An exponential integrator is applied to time-discretize the equations of motion. An observability analysis is performed and the observability conditions for the unknown input forces are derived. Furthermore, it is shown that no position-level measurements are required if only the forces are of interest. The proposed approach is numerically validated and compared to the current general approach. The validation demonstrates that the proposed approach provides superior estimation results.

Keywords: Kalman filter, Input estimation, Observability, Floating frame of reference, Model reduction.

1 INTRODUCTION
The forces acting on a vehicle are an important area in vehicle dynamics research. Together with the vehicle state, they are essential for determining the dynamic behavior of the vehicle. Accurate knowledge of the external forces, mostly tyre forces, is of specific interest during the vehicle design phase, and is of great value for vehicle control systems that attempt to improve vehicle handling and safety. A complete measurement of the forces and states of interest, however, requires an expensive and complicated sensor installation which is infeasible for mass-produced vehicles. This has motivated the use of model-based estimation techniques such as the Kalman filter [1] and its many variations [2]. The main strength of the Kalman filter is that it can optimally combine the a-priori knowledge of a system, by the use of a model, and the measurements.

The regular Kalman filter assumes that the external inputs to the system are known, but in vehicle applications this is not the case: the external tyre forces are unknown. In order to perform state estimation in the presence of unknown inputs so-called unbiased Kalman filters have been proposed [3, 4]. However, in vehicle applications it is of special interest to obtain an estimate of the unknown inputs to the system as well. Several variations of the regular Kalman filter exist that can perform a combined estimation of the states and inputs. In this work the augmented Kalman filter is chosen [5]. This approach adds the unknown inputs to the state vector and treats them as additional states to be estimated. A model for the unknown inputs has to be provided.
In order to obtain good estimation results a sufficiently accurate system model is required. Currently the general approach is still to use a simplified ad-hoc model for the Kalman filter. Of course, a large model-plant mismatch will lead to inaccurate estimation results. Furthermore, also observability problems can occur if the model is too much simplified and multiple unknown inputs are present. This type of problem occurs in vehicle dynamics applications. Consider a simplified vehicle dynamics model such as the rigid four-wheel vehicle model that is often used for estimation purposes [6, 7, 8]. There are eight unknown input forces that need to be estimated: a longitudinal and a lateral tyre force at each wheel. An observability analysis shows that employing the augmented Kalman filter to estimate the individual tyre forces leads to an unobservable system. This is caused by the fact that the four-wheel vehicle model does not contain an effect that allows to differentiate between the contributions of all the different forces. It is therefore impossible to estimate the individual tyre forces. In literature this issue is typically circumvented by making assumptions about the input forces such as a known tyre force distribution or by assuming a tyre force model with known parameters. This reduces the number of independent variables that need to be estimated such that an observable system can be obtained. However, it is clear that these assumptions reduce the robustness and the accuracy of the estimation because this information is generally not available.

Therefore, this work proposes a coupled state/input estimation approach that allows to estimate multiple unknown input forces without the need of any assumptions about these forces. Instead of using a rigid vehicle model in the Kalman filter, it is proposed to take the flexibility of the vehicle body into account. This adds an effect to the model that allows to differentiate between the contributions of all the different forces. The required information can be extracted from additional measurements, such as strain measurements. As a result an observable system can be obtained. A similar approach is followed in structural dynamics applications to estimate external forces [9, 10, 11]. The difference is that in this work the deformable body can also undergo large translational and rotational displacements. This results in a nonlinear model. Section 2 discusses the derivation of this model, in a floating frame of reference formulation [12]. This section also discusses the use of a reduced order finite element model to describe the flexible behavior of the vehicle body. Section 3 introduces the discrete-time augmented extended Kalman filter and discusses the required derivatives of the model. In this section also an observability analysis is performed and the results are discussed. Finally in Section 4 the proposed approach is numerically validated and compared to the current general approach.

2 REDUCED ORDER MODEL FOR STATE/INPUT ESTIMATION

The equation of motion of a deformable body that undergoes large translational and rotational displacements in a floating frame of reference formulation is [12]:

$$M(q)\ddot{q} + C\dot{q} + Kq + Q_v(q, \dot{q}) = Q_{ext}(q) \quad (1)$$

In this equation \(q\) is the vector of time-dependent generalized coordinates of the system, and \(M, C\) and \(K\) are respectively the mass-, damping- and stiffness matrix. \(Q_v\) is a quadratic velocity vector that contains the gyroscopic and Coriolis force components and \(Q_{ext}\) is the vector of generalized external forces. In the floating frame of reference formulation the motion of the body is separated into a large motion of the body reference frame and a linear elastic deformation with respect to the reference frame. The configuration of the deformable body is then defined by a coupled set of generalized reference and elastic coordinates:

$$q = \begin{bmatrix} R \\ \theta \\ \mathbf{q}_f \end{bmatrix} \quad (2)$$
In which $R$ and $\theta$ are the generalized reference coordinates and $q_f$ is the vector of generalized elastic coordinates. $R$ is the global position of the origin of the body reference frame and $\theta$ is a set of $n_r$ rotational coordinates that describes the orientation of the body reference frame with respect to the global frame. In the remainder of the text only the planar analysis is discussed for the sake of simplicity, but the presented results can easily be extended to the spatial case. In the planar case $n_r$ is equal to one, and $\theta$ can thus be written as $\theta$.

Since this work focuses on input force estimation, only the derivation of $Q_{ext}$ is provided. For the derivation of the full model the reader is referred to [12].

Assume that an external force $f_{ext}$ is applied at a point $P$ of the deformable body. The global position of point $P$ can be expressed in function of the generalized coordinates as:

$$r_p = R + A(\theta)(\tilde{u}_0 + N_p q_f)$$  \hspace{1cm} (3)

where $\tilde{u}_0$ is the local position vector of point $P$ in the undeformed state and the term $N_p q_f$ describes the local linear elastic deformation. $N_p$ is a partition of a shape matrix that can be obtained by using e.g. the finite element method. $A$ is a transformation matrix that transforms the local vector coordinates to the global frame. The generalized external force vector $Q_{ext}$ can be obtained as the product of the external force and the partial derivative of the global position $r_p$ of the point of application with respect to the generalized coordinates:

$$Q_{ext}(q_f) = L^T(q_f) f_{ext} = \begin{bmatrix} \partial A(\theta)(\tilde{u}_0 + N_p q_f) \\ A(\theta) N_p \end{bmatrix}^T \begin{bmatrix} f_x \\ f_y \end{bmatrix}_{ext}$$  \hspace{1cm} (4)

In which the matrix $L$ describes how the external force loads the generalized coordinates. For estimation purposes, however, it makes more sense to express the external forces with respect to the body reference frame. This is achieved by substituting in (4) the external forces by the product of the transformation matrix $A$ and the external forces expressed with respect to the body reference frame:

$$Q_{ext}(q_f) = \tilde{L}^T(q_f) \tilde{f}_{ext} = \begin{bmatrix} A^T(\theta)(\tilde{u}_0 + N_p q_f) \\ N_p \end{bmatrix}^T \begin{bmatrix} \tilde{f}_x \\ \tilde{f}_y \end{bmatrix}_{ext}$$  \hspace{1cm} (5)

where the tilde stands for the skew symmetric transformation.

The remainder of this section consists of three subsections. The first subsection discusses the use of a reduced order finite element model to describe the behavior of the deformable body. The second subsection discusses a technique called state augmentation, which allows to obtain estimates of the unknown external forces in a Kalman filtering approach. The third subsection discusses the time-discretization of the equations of motion, which is required to employ the discrete-time extended Kalman filter.

### 2.1 Model order reduction by modal truncation augmentation

The flexible behavior of the body can be modeled using the finite element method. However, if the goal is to perform the model-based estimation in an online application, it is typically not feasible to use a full finite element model. This is due to the large amount of degrees-of-freedom (DOFs) and the high-frequency content of these models. Therefore, a reduced order model should be employed. Over the past decades, several model order reduction techniques have been proposed [13].

The aim of this work is on structural systems, which are second-order systems. The most widely used reduction technique for structural systems is modal reduction, which is based on a truncated modal expansion of the system. However, this technique exhibits bad convergence because the modes retained are usually not sufficient to represent the spatial distribution of the applied load. This shortcoming can be overcome by adding to the set of retained modes additional component modes that take the specific loading situation into account. This technique is applied in this work.
to reduce a finite element model of the body. A statically complete dynamic mode superset is employed consisting of the rigid-body modes \( \Psi_{rb} \), a set of retained free-free normal modes \( \Psi_{ff} \), and a set of residual inertia-relief attachment modes \( \Psi_{ra} \) \[14\]:

\[
\Psi = [\Psi_{rb} \, \Psi_{ff} \, \Psi_{ra}] \tag{6}
\]

The reduction of the finite element model is performed by representing the component’s physical displacement DOFs \( d \), in terms of the generalized coordinates \( p \):

\[
d = \Psi p \tag{7}
\]

In which the number of modes in \( \Psi \) is typically much less than the number of DOFs of the original system. The resulting reduced finite element model is then used to construct the model in (1) according to the floating frame of reference formulation.

2.2 State augmentation for input estimation

In order to use model (1) in a Kalman filtering approach, it has to be transformed into first-order form:

\[
\dot{x} = g(x) \tag{8}
\]

The state-space vector for this system becomes:

\[
x = \begin{bmatrix} q \\ \dot{q} \end{bmatrix} \tag{9}
\]

and the corresponding continuous-time system equations are:

\[
\begin{bmatrix} \dot{q} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} 0 & I \\ -M^{-1}K & -M^{-1}C \end{bmatrix} \begin{bmatrix} q \\ \dot{q} \end{bmatrix} + \begin{bmatrix} 0 \\ M^{-1}L^T \end{bmatrix} f_{ext} - \begin{bmatrix} 0 \\ M^{-1}Qv \end{bmatrix} \tag{10}
\]

where for ease of notation, the dependency of \( M, L \) and \( Qv \) on \( q \) and \( \dot{q} \) is not shown. These equations can be written as:

\[
\dot{x} = A(x)x + B(x)\hat{f}_{ext} - c(x) \tag{11}
\]

For estimating the input simultaneously with the system state in a Kalman filtering approach, a technique called state augmentation is employed \[5\]. In this approach the unknown input forces are added to the state vector and are treated as additional states to be estimated. This leads to the augmented state vector \( x^* \):

\[
x^* = \begin{bmatrix} x \\ \hat{f}_{ext} \end{bmatrix} \tag{12}
\]

A model for the unknown forces has to be provided. Here, a zeroth-order random walk model is employed for each unknown force. This model assumes the force to be constant except for an additive uncertainty:

\[
\dot{\hat{f}}_{ext} = 0 + \hat{r}_f \tag{13}
\]

where \( \hat{r}_f \) is continuous-time noise, which indicates that the rate of change of the external force is expected to be a random process. This approach has been applied with success by Naets \[9, 10, 15\].
and Lourens [11] and higher order versions of this approach have been used by Ray [6, 7]. In general, $r_f$ is not exactly known and serves as a tuning parameter for the Kalman filter. Since the external forces can show strong variations, $r_f$ is typically quite high. The augmented continuous-time system model becomes:

$$\dot{x} = A(x) \dot{x} - c(x) + \begin{bmatrix} 0 \\ r_f \end{bmatrix}$$

which can be written as:

$$\dot{x}^* = A_c(x^*) \dot{x}^* - c_c(x^*) + r_{x^*}$$

where $r_{x^*}$ is the continuous-time process noise vector with covariance matrix $Q_c$. Notice that the noise on the continuous-time system model is assumed to be zero and all noise is assumed on the unknown forces. Furthermore, it is also important to mention that this system is nonlinear. Both the matrix $A_c^*$ and the vector $c_c^*$ are dependent on the system state.

### 2.3 Discretization of equations of motion

In this work the discrete-time extended Kalman filter is employed because it is particularly suitable for efficient computer implementation. In order to apply this filter to system (16), the equations of motion need to be time-discretized. Several integration schemes exist to discretize these equations. In this work an exponential integration scheme is chosen with zero-order hold for the inputs [16]. This approach allows for larger time steps than typical explicit integrators. The following discrete-time equations of motion are obtained:

$$\begin{bmatrix} x \\ \dot{f}_{ext} \end{bmatrix}_k = \begin{bmatrix} e^{A \Delta t} & A^{-1} (e^{A \Delta t} - I) B \\ 0 & I \end{bmatrix} \begin{bmatrix} x \\ \dot{f}_{ext} \end{bmatrix}_{k-1} - \begin{bmatrix} A^{-1} (e^{A \Delta t} - I) c \\ 0 \end{bmatrix}_{k-1} + \begin{bmatrix} 0 \\ r_f \end{bmatrix}_{k-1}$$

where for ease of notation, the dependency of $A$, $B$ and $c$ on $x$ is not shown. These equations can be written as:

$$x_k^* = A_d(x^*) x_{k-1}^* - c_d(x^*) + r_{x_{k-1}}$$

$$= g_d(x_{k-1}^*) + r_{x_{k-1}}$$

where $\Delta t$ is the time step and $r_{x_{k-1}}$ is the discrete-time process noise vector with covariance matrix $Q_d$. The discrete-time covariance matrix is calculated as:

$$Q_d = \int_0^{\Delta t} e^{A(x^*)^\top} Q_c e^{A(x^*)^\top} \, dt$$

Solving this integral is not straightforward, however, $Q_d$ can also be calculated by first constructing a matrix, and computing the matrix exponential of it [17]:

$$M(x^*) = \begin{bmatrix} A_c^T(x^*) & Q_c \\ 0 & A_c(x^*) \end{bmatrix}$$

$$N(x^*) = e^{M(x^*) \Delta t} = \begin{bmatrix} \cdots & A_d^{-1}(x^*) Q_d \\ 0 & A_c^T(x^*) \end{bmatrix}$$
The covariance matrix $Q_d$ is then evaluated as the product of the transpose of the lower-right partition of $N$ and the upper-right partition of $N$:

$$Q_d = (A_d^T(x^*))^T (A_d^{-1}(x^*)Q_d)$$  \hspace{1cm} (23)$$

The discrete-time equations of motion (19) and the covariance matrix (22) can be inserted in the discrete-time extended Kalman filter to perform the coupled state/input estimation.

3 THE AUGMENTED EXTENDED KALMAN FILTER FOR COUPLED STATE/INPUT ESTIMATION

As mentioned before, the system considered is nonlinear. In this work, the extended Kalman filter is selected to perform the estimation. This filter is a straightforward extension of the original linear Kalman filter to nonlinear systems [2].

The system equations of motion (19) are complemented by the (nonlinear) measurement equations:

$$y_k = h(x^*_k) + r_y$$  \hspace{1cm} (24)$$

in which $y_k$ contains the sensor measurements as obtained by the (nonlinear) measurement equations $h$ and the measurement noise $r_y$. The noise is assumed to be Gaussian, white and centered, and has covariance matrix $R$.

Just like the original linear Kalman filter, the extended Kalman filter algorithm is recursive and operates in two steps: a time update step and a measurement update step. The time update step predicts the current system state $\hat{x}^{+ -}_k$ and propagates the state error covariance matrix $P^{-}_k$:

$$\hat{x}^{+ -}_k = g_d(\hat{x}^{+ -}_{k-1})$$  \hspace{1cm} (25)$$

$$P^{-}_k = GP^{-}_{k-1}G^T + Q_d$$  \hspace{1cm} (26)$$

Once the measurement $y_k$ is available, the Kalman gain $K_k$ is calculated and is used to calculate the updated system state estimate $\hat{x}^{+ +}_k$ by incorporating the information available in the measurement. The updated state error covariance matrix $P^{+}_k$ is also computed:

$$K_k = P^{+ -}_kH^T(HP^{+ -}_kH^T + R)^{-1}$$  \hspace{1cm} (27)$$

$$\hat{x}^{+ +}_k = \hat{x}^{+ -}_k + K_k(y_k - h(\hat{x}^{+ -}_k))$$  \hspace{1cm} (28)$$

$$P^{+}_k = (I - K_kH)P^{+ -}_k$$  \hspace{1cm} (29)$$

In both the time- and measurement update step, the propagation of the state error covariance is performed by linearizing, respectively, the system- and measurement equations around the current configuration:

$$G = \frac{\partial g_d(x^*_k)}{\partial x^*}$$  \hspace{1cm} (30)$$

$$H = \frac{\partial h(x^*_k)}{\partial x^*}$$  \hspace{1cm} (31)$$

Calculating $H$ is usually rather straightforward. Calculating $G$, however, the derivative of the equations of motion, is more difficult. This derivative is elaborated in the next subsection.
3.1 Required derivatives

In order to employ the extended Kalman filter, the derivative of the discrete-time equations of motion with respect to the augmented state vector is required (30). This derivative is a square matrix in which each column \( j \) is the derivative of the equations of motion with respect to augmented state \( j \):

\[
G_{c,j} = A_{d,j} (x_{k-1}^*) + \frac{\partial}{\partial x_j} \left( \begin{bmatrix} e^{A\Delta t} & A^{-1} (e^{A\Delta t} - I) \end{bmatrix} B \right) x^* - \left( \begin{bmatrix} A^{-1} (e^{A\Delta t} - I) \end{bmatrix} c \end{bmatrix} \right)_{k-1}
\]

for \( j = 1 \ldots n^* \), where \( n^* \) is the number of augmented states. For ease of notation, the dependency of \( A, B \) and \( c \) on \( x \) is not shown. In order to evaluate (32) the following expressions must be used:

\[
\frac{\partial A}{\partial x_j} = \begin{bmatrix} 0 & 0 \\ M^{-1} \frac{\partial M}{\partial x_j} M^{-1} K & M^{-1} \frac{\partial M}{\partial x_j} M^{-1} C \end{bmatrix}
\]

\[
\frac{\partial B}{\partial x_j} = \begin{bmatrix} 0 \\ M^{-1} \frac{\partial \mathbf{L}^T}{\partial x_j} - M^{-1} \frac{\partial M}{\partial x_j} M^{-1} \mathbf{L}^T \end{bmatrix}
\]

\[
\frac{\partial c}{\partial x_j} = \begin{bmatrix} 0 \\ M^{-1} \frac{\partial \mathbf{Q}_v}{\partial x_j} - M^{-1} \frac{\partial M}{\partial x_j} M^{-1} \mathbf{Q}_v \end{bmatrix}
\]

For the derivative of the mass matrix \( M \) and of the quadratic velocity vector \( Q_v \) the reader is referred to [12]. The derivative of matrix \( \mathbf{L} \) can be obtained rather straightforward from (5).

3.2 Observability

The observability is a very important property of the estimator. Therefore, before moving on to the numerical validation, an observability analysis is performed. A system is said to be observable if its state vector at a certain time instant can be uniquely determined given a finite sequence of its outputs. Less formally, observability means that the measurements can provide sufficient information for estimating the state vector. Observability is thus a property of a certain system-measurement combination. In this work four different kinds of measurements are considered:

- position measurement, using e.g. an optical tracking system,
- acceleration measurement, using accelerometers,
- angular velocity measurement, using a gyroscope,
- strain measurement, using strain gauges.

As mentioned before, the model considered is nonlinear. Determining the observability for a nonlinear system is substantially more difficult than for a linear system. An approximate method of checking the observability of nonlinear systems is to perform a linearized observability analysis. This can already provide some general conditions for observability. In this work the Popov-Belevitch-Hautus (PBH) test of observability of a linear system is applied on the linearized continuous-time system [18]. This test states that the linearized system is observable if and only if:

\[
\text{rank} \left[ \begin{bmatrix} G_c - sI \\ H \end{bmatrix} \right] = n^*, \text{ for all } s \in \mathbb{C}
\]
where $G_c$ is the derivative of the continuous-time system and $H$ is the derivative of the measurement equations. It is sufficient to check the rank for values of $s$ equal to the eigenvalues of the matrix $G_c$ since for all other values of $s$ the matrix $G_c - sI$ is full rank by definition of the eigenvalue-problem. Here, only the case $s = 0$ is investigated because this is where the problems typically occur for force estimation [9]. Careful study of the PBH-matrix (36) shows that the conditions for observability of the complete state vector in this case are:

- at least one global $x$- and $y$-position measurement is performed,
- a certain minimum amount of strain measurements is performed.

Further analysis of the PBH-matrix (36) and of the model considered reveals that it is possible to observe the external forces by performing only the strain measurements. In this case, the states $R$, $\dot{R}$ and $\theta$ are unobservable but in this work the main interest is in the input forces such that these variables are not of specific interest. This is a very good feature since position measurements are difficult to obtain in practice. The conditions for observability of the external forces are:

- the number of DOFs of the reduced model is larger than or equal to the number of forces,
- the number of strain measurements is larger than or equal to the number of forces.

These conditions indicate that not only accuracy considerations can dictate the size of the reduced order model, but also the observability requirements have to be taken into account.

It is interesting to perform a similar observability analysis if the body is modeled as rigid. This corresponds to the current general approach. In this case, the information available in the strain measurements cannot be extracted because the model does not contain this effect. The analysis shows that it is only possible to estimate a maximum of three unknown external forces, requiring at least an angular velocity measurement, and a longitudinal and a lateral acceleration measurement.

4 NUMERICAL VALIDATION

In order to validate the proposed approach, a numerical experiment is performed in MATLAB. The proposed approach is compared to the current general approach where the flexibility of the body is not taken into account in the model for the Kalman filter. The first subsection introduces the system and describes the models used. The second subsection shows and discusses the simulation results.

4.1 Model description

The system shown in Figure 1 is used to validate the proposed estimation approach. It consists of a flexible vehicle body suspended by a vertical suspension system. The vehicle is loaded by a known gravitational force, the reaction forces due to the irregular road profile and an aerodynamic force. In total there are six unknown external forces that need to be estimated, as shown in Figure 1. Table 1 summarizes the main properties of the vehicle body, Table 2 shows the properties of the spring-damper suspension system.

The model for the Kalman filter is derived according the floating frame of reference formulation. The model of the vehicle body is based on a finite element mesh. Due to the large amount of DOFs and the high-frequency content of this model, it is not feasible to use the full finite element model in the Kalman filter. Therefore, a model reduction performed. The reduction technique outlined in subsection 2.1 is applied to the full finite element model. The reduction is performed such that in the frequency range of the applied loads a good approximation of the full finite element model is obtained. The number of elastic DOFs is reduced from 8726 to 12.
Eleven measurements are performed: ten strain measurements and a measurement of the angular velocity of the vehicle body. The noise on the measurements is assumed to be Gaussian white noise with realistic values for the sensors.

This system-measurement combination leads to a partially observable system. The external forces are observable, but the states $R$, $\dot{R}$ and $\theta$ are not. This is because no absolute position measurement is performed. The sampling frequency of the estimator is 1kHz.

The measurements and the true force trajectories are generated using the reference model. The reference model is also derived according to the floating frame of reference formulation. A high-accuracy Krylov reduced finite element model of the vehicle body is employed [19] of dimension 64. Convergence tests were performed to ascertain that this reduced model is a very good approximation of the full finite element model in a wide frequency range. The equations of motion are integrated with a generalized-$\alpha$ solver with time step 1ms and spectral radius $\rho_\infty = 1$ [20].

The proposed approach, as described above, is compared to the current general approach where the vehicle body is assumed to be rigid in the model for the Kalman filter. In this case, three measurements are performed: a longitudinal and a lateral acceleration measurement, and an angular velocity measurement. The acceleration measurements provide accelerations in the frame attached to the vehicle body.

### 4.2 Simulation results

For the numerical experiment, an imposed displacement is applied to both wheels and a concentrated aerodynamic force is applied at the rear of the vehicle. The imposed displacement is the superposition of a sine sweep (0 to 3Hz) and a ramp. A time-delay of 0.5s exists between the excitation at the front and the rear wheel. The aerodynamic force is the superposition of a sine at constant frequency (3Hz), and a constant load. In total there are six unknown external forces that need to be estimated, as shown in Figure 1.
Figure 2. Estimation results of the six unknown external forces: (a) force estimates, (b) error variance of the force estimates.

Figure 2 shows the simulation results. Part (a) of the figure clearly shows that the proposed approach is able to deliver accurate force estimates. At first sight, the accuracy of the estimate for $F_{x,a}$ seems rather low, but notice the low amplitude of this force as compared to the other forces. If the rigid approach is employed the force estimates are significantly worse. This is not caused by the model-plant mismatch, but by the fact that in this case the system-measurement combination is unobservable. The three measurements do not provide sufficient information to estimate the six external forces. Performing any additional measurements, such as strain measurements, will not improve the observability because the rigid body model does not contain these effects. This demonstrates that in order to obtain an observable system a model with a certain complexity is required.

Part (b) of Figure 2 shows the evolution of the force estimate error variances. These results confirm the previous statements. The proposed approach leads to observability of the forces: the error variances quickly converge. The rigid approach leads to unobservability of the forces: the error variances diverge. Over time the estimates obtained with the rigid approach will further deteriorate.
5 CONCLUSIONS

In this work the discrete-time augmented extended Kalman filter is employed to perform a combined estimation of the current vehicle state and the input forces, such as the tyre forces. Currently, the general approach is still to use a simplified ad-hoc model for the Kalman filter. In vehicle dynamics applications, this leads to an unobservable system which makes it impossible to estimate the unknown forces unless a-priori assumptions about these forces are made. These assumption reduce the robustness and the accuracy of the estimation results. In order to solve this issue, this work proposed to take the flexibility of the vehicle body into account in the model for the Kalman filter. This adds an effect to the model that allows to differentiate between the contributions of all the different input forces. The required information can be extracted from additional measurements, such as strain measurements. As a result an observable system is obtained without the need of any assumptions about the unknown forces. The model is derived according to the floating frame of reference formulation and a reduced order model finite element model is employed to describe the flexible behavior of the vehicle body. An exponential integration scheme is applied to time-discretize the equations of motion. An observability analysis was performed and the observability conditions for the unknown input forces were derived. They state that the unknown forces can be observed if and only if the number of degrees-of-freedom of the reduced order model is larger than or equal to the number of unknown forces, and if the number of strain measurements is larger than or equal to the number of unknown forces. Furthermore, it was shown that no position-level measurements are required if only the forces are of interest. The proposed approach is numerically validated and is compared to the current general approach where the flexibility of the body is not taken into account in the model for the Kalman filter. The validation demonstrates that the proposed approach provides superior estimation results.

ACKNOWLEDGEMENTS

This work has been partially supported by the Flanders’ DRIVE project ‘GPS-positioning’, granted by the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT-Vlaanderen). This work benefits from the Belgian Programme on Interuniversity Attraction Poles, initiated by the Belgian Federal Science Policy Office (DYSCO). The Fund for Scientific Research – Flanders (F.W.O.), and the Research Fund KU Leuven are also gratefully acknowledged for their support.

REFERENCES


Verification and Optimization Advantage of Inerter Devices Apply to Grounded Vehicle Dynamics

Thanh-Tung TRAN*, Hiroshi HASEGAWA#

Graduate of Systems Engineering and Science
Shibaura Institute of Technology, Japan
307 Fukasaku, Minuma-ku, Saitama-shi 337-8570, Japan
*nb12507@sic.shibaura-it.ac.jp, #h-hase@sic.shibaura-it.ac.jp

ABSTRACT
The suspension system needs to be soft to insulate against road disturbances and hard to insulate against load disturbances. It cannot achieve with a traditional passive suspension that only considered to the stiffness and damper. In this paper, we propose some inerter designs, which have some advantages for suspension system by improving vehicle oscillation. Thus, we developed the verification process to numerically calculate the suspension model for quarter-car model. We applied the procedure to the linear quarter-car model with conceptual inerter model employment. The mathematical model figure out with assumption of simple theories as constant inertance was defined compare with the computational model was obtained after implementing the detail factors and coefficients inside the inerter mechanism. We optimize design of model based on the minimization of cost functions for displacement, tire deflection with constraint function of suspension deflection limitation and the energy consumed by the inerter. The paper clarifies some issues related to suspension system with inerter to reduce sprung mass displacement and tire deflection in quarter-car model. The advantage of research is verify and optimal a new mechanism, the inerter; this system can improve the vehicle oscillation on quarter-car model with different parameters that shows the benefit of the inerter in proposal suspension system.

Keywords: Inerter, Verification, Optimization, Suspension, Dynamics.

1 INTRODUCTION
Passive, semi-active and active suspension systems have been utilized to improve ride comfort of vehicles and their effectiveness has also been demonstrated. To achieve it, several control methods have been proposed, but most of them relate the active suspension [1] [2] [3]. To developing suspension system, it relates some specifications as follow.

1.1 Comfort Specifications:
The comfort characteristics are considered, it mainly related to displacement, chassis vibration, noise, etc. It has an impact on the driver reaction time, accuracy, situation evaluation and decision abilities, which makes this objective particularly active in the automotive community. According to these kind of models, they indicate some sensitive frequency zones related to the heart, the head, etc. having resonance or gain amplifications around some specific frequencies according to different disturbances such as the road irregularities [4].

In this paper, the comfort is not directly discussed, but evaluated through the chassis analysis as sprung mass displacement. The comfort feeling analysis is performed by analyzing some specific frequencies of the vertical behavior of the quarter-vehicle model. We focus on the analysis of simpler variables behavior with respect to road uneveness, such as vertical acceleration $\ddot{Z_s}$ and displacement $Z_s$ of the chassis. Then an improvement on these variables will imply comfort improvement with mathematical objective is:
\[ \min Z_s(t) := \min Z_u(t) \] (1)

1.2 Road-Holding Specifications:
Road-holding is a vehicle property which characterizes the ability of the vehicle to keep contact with the road and maximize wheel tracking to road unevenness. It is very simplified manner, the longitudinal \((F_{lx})\) lateral \((F_{ly})\) forces of each tire as follows:

\[ F_{lx} = F_n \mu_x; \quad F_{ly} = F_n \mu_y \] (2)

Where \(\mu_x\) and \(\mu_y\) are the nonlinear functions, dependent on the slip ratio, the slip angle and the road roughness characteristics.

These forces are also affine functions of \(F_n\), the normal load, defined as:

\[ F_n = (M_s + M_u)g - k_t (Z_u - Z_r) \] (3)

Where \(M_s\) and \(M_u\) are the sprung and unsprung masses, \(k_t\) is the vertical tire stiffness characteristic and tire deflection \(Z_{t-def} = Z_u - Z_r\).

Because \((M_s + M_u)g > 0\) and \(k_t > 0\) then: \(F_n\) max when \(tire\) deflection \(Z_{t-def}(t) = Z_u(t) - Z_r(t)\) go to minimization.

1.3 Suspension Limitations:
When evaluating a suspension system and its associated control algorithm, to take into account the static suspension stroke limitations \(Z_{sus} = Z_s - Z_u\) which should always remain in between the limitations defined by the technology, i.e. then:

\[ \min Z_{sus} \leq Z_{sus} \leq \max Z_{sus} \] (4)

Where \(\min Z_{sus}\) an \(\max Z_{sus}\) are the suspensions deflection limits. This constraint is very important for practical applications, in order to preserve the mechanical suspension system.

1.4 Conceptual Inerter Model
The conceptual model is defined shown in Figure 1, representation of the solid mechanics behavior of the reality of interest. The model include those mechanisms that impact the key mechanical and physical processes that will be of interest for the intended use of the model. The activity of conceptual model development involves formulating a mechanics-based representation of the reality of interest that is amenable to mathematical and computational modeling and that is expected to produce results with adequate accuracy for the intended use.

The formulation of the conceptual model is important to the overall model development process because many fundamental assumptions are made that influence interpretation of the simulation.
results. These assumptions include the determination of how many separate parts or components will be included in the model, the approach to modeling the material behavior, the elimination of unimportant detail features in the geometry, and the selection of interface and boundary types, e.g., fixed, pinned, contact, friction, etc. If an important mechanical phenomenon is omitted from the conceptual model, the resulting simulations might not be adequate for the intended use of the model.

In previous research, an inerter was defined as a mechanical two-terminal, one-port device with the property that the equal and opposite force applied at the nodes is proportional to the relative acceleration between the nodes through a rack, pinion, and gears [6].

To approximately model the dynamics of the device of Figure 1, let \( r_1 \) be the radius of the rack pinion, \( r_2 \) the radius of the gear wheel, \( r_3 \) the radius of the flywheel pinion, \( \gamma \) the radius of gyration of the flywheel, \( m \) the mass of the flywheel.

The following relation holds:

\[
F = b(\dot{v}_2 - \dot{v}_1) \tag{5}
\]

The constant of proportionality \( b \) is called the inertance,

And has units of kilograms:

\[
b = ma_1^2 \omega_2^2 \tag{6}
\]

Where \( a_1 = \gamma/r_3 \) and \( a_2 = r_2/r_1 \).

It stored energy equal:

\[
E = \frac{1}{2}b(v_2 - v_1)^2 \tag{7}
\]

2 MATHEMATICAL MODEL

The development of the mathematical model consists of specifying the mathematical descriptions of the mechanics represented in the conceptual model. In the mathematical model, principles of mechanics, the material behavior, interface properties, loads, and boundary conditions are cast into equations and mathematical statements. The specification of the mathematical model then allows the model input parameters to be defined. The model input parameters describe the various user-specified inputs to the model, such as material constants, road disturbance, and friction coefficient are fixed. The domain of interest can then be expressed in terms of these parameters.

Therefore, in mathematical model, \( b \) value is constant, we do not define for inerter component as gear inside. Thus we assumption that inerter ignore all coefficient of fraction inside, we only define for quarter-car suspension as follow. For the quarter-car model, the suspension admittance function which relates the suspension force to the strut velocity through spring, damper and inerter in parallel [7].

The module of sprung mass body was represented by these equations:

\[
M\ddot{Z}_2(t) = F_k(t) + F_c(t) + F_b(t) \tag{8}
\]

Where:

\[
F_k(t) = k(Z_1(t) - Z_2(t))
\]

\[
F_c(t) = c\left(\dot{Z}_1(t) - \dot{Z}_2(t)\right) \tag{9}
\]

\[
F_b(t) = b(\ddot{Z}_1(t) - \ddot{Z}_2(t))
\]

The un-sprung mass module equation:
\[ m\dddot{Z}_1(t) = F_{kt}(t) - (F_k(t) + F_c(t) + F_b(t)) \]  
\[ (10) \]

Where: \( F_{kt}(t) = k_t(Z_0(t) - Z_1(t)) \)

The equations of motion in the Laplace transformed domains are:

\[ M_s s^2 \ddot{Z}_s = -sQ(s)(\dddot{Z}_s - \dddot{Z}_u) \]
\[ M_u s^2 \ddot{Z}_u = sQ(s)\dddot{Z}_s - sQ(s)\dddot{Z}_u + k_t(\dddot{Z}_r - \dddot{Z}_u) \]  
\[ (11) \]

Where: \( k, c, b, k_t \) is stiffness, damping, inertence and tire stiffness in respectively.

\[ Q(s) = \frac{k}{s} + c + bs \] , present for Laplace transform of suspension system employing inerter in parallel structure.

We integrated inerter mechanism that applied on passive suspension system. First, we use conventional suspension model with initial parameter from formula car. Second, base on mathematical model, we apply inerter on suspension system in parallel structure show on Figure 2. Then we focus to analyze the effects of inerter on the system as displacement of sprung mass and tire deflection.

Furthermore, this mathematical model used fixed initial conditions such as \( b \) value under mass unit which there is no parameter present for factor and coefficients inside. Changing the \( b \) value means change prototype of inerter mechanism in respectively. Thus mathematical model is the simple model then we try to verify that it can be work same the real one correctly.

![Figure 2. The Laplace transform, conventional and parallel mathematical of quarter-car model in respectively.](image)

### 3 COMPUTATIONAL MODEL

The computational model is the numerical implementation of the mathematical model that will be solved on a computer to yield the computational predictions (simulation results) of the system response. As defined herein, the computational model includes the type and degree of spatial discretization of the component inside, the temporal discretization of the governing equations, the solution algorithms to be used to solve the governing equations, and many coefficients criteria for the numerical solutions.
The computational model can be simple or complicated. Though an analyst might be tempted to jump directly from a component description of the reality of interest to the development of a computational model, it is valuable to think through the process of conceptual modeling, mathematical modeling, and computational modeling to have a thorough understanding of what assumptions and mathematical simplifications underlie the computational model.

For the last model, we design computational quarter-car model with more detail of inerter components [7] in gear, rack, pinion, flywheel and friction parameter show in Figure 3. Each of them has difference characteristics on working as frictions or efficient rate so we can estimate from simulation results. Furthermore, we integrated inerter mechanism which applied on passive suspension system. For each model, we have to calibrate to closing optimization inertance value. Each of them has difference characteristics on working as frictions or efficient rate so they can effect to simulation results.

![Figure 3. The computational parallel suspension on quarter-car model.](image)

Commonly, at some stage of validation, the modeler will find that the computational model needs revisions to achieve the desired accuracy or to account for new requirements. In a general sense, there are two classes of possible revisions to the mathematical and computational models. The first class of revisions covers updates to parameters in the mathematical or computational model that are determined by calibrating the computational model to experimental data, e.g., additional component parameters, modal damping coefficients for linear vibration, or friction coefficients for a mechanical interface. The second class of revisions covers changes to the form of the mathematical or conceptual model to improve the description of the mechanics of interest so that better agreement with the reference computational data can be achieved. The classes of gear inerter parameter are discussed below Table 1,2,3.

### Table 1. Gear component parameters index.

<table>
<thead>
<tr>
<th>Title</th>
<th>Variable name</th>
<th>Unit</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Radius of gear wheel</td>
<td>$r_2$</td>
<td>mm</td>
<td>50</td>
</tr>
</tbody>
</table>
Radius of flywheel pinion | $r_3$ | mm | 10
Width of tooth face | $b$ | mm | 15
Immersed gear height | $h$ | mm | 0
Poisson's ratio | pois | null | 0.3
Young's modulus | $E_Y$ | N/m$^2$ | $2.1 \times 10^{11}$
Coefficient of viscous friction | vis | Nm/(rev/min) | 0

Table 2. Helical rack and pinion component parameters index.

<table>
<thead>
<tr>
<th>Title</th>
<th>Variable name</th>
<th>Unit</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Radius of the pinion</td>
<td>$r_1$</td>
<td>mm</td>
<td>10</td>
</tr>
<tr>
<td>Helix angle</td>
<td>alpha</td>
<td>degree</td>
<td>5</td>
</tr>
<tr>
<td>Pinion axis angle</td>
<td>beta</td>
<td>degree</td>
<td>0</td>
</tr>
<tr>
<td>Contact stiffness</td>
<td>$k$</td>
<td>N/m</td>
<td>$1 \times 10^9$</td>
</tr>
<tr>
<td>Contact damping</td>
<td>damp</td>
<td>N/(m/s)</td>
<td>$1 \times 10^6$</td>
</tr>
<tr>
<td>Static coefficient</td>
<td>must</td>
<td>null</td>
<td>0.12</td>
</tr>
<tr>
<td>Friction coefficient</td>
<td>musl</td>
<td>null</td>
<td>0.1</td>
</tr>
<tr>
<td>Stribeck constant</td>
<td>astrib</td>
<td>m/s</td>
<td>0.001</td>
</tr>
<tr>
<td>Stick displacement threshold</td>
<td>dtrel</td>
<td>mm</td>
<td>0.001</td>
</tr>
<tr>
<td>Equivalent viscous friction during stiction</td>
<td>rtors</td>
<td>N/(m/s)</td>
<td>100000</td>
</tr>
</tbody>
</table>

Table 3. Rotary parameter

<table>
<thead>
<tr>
<th>Title</th>
<th>Variable name</th>
<th>Unit</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moment of inertia</td>
<td>I</td>
<td>g*m$^2$</td>
<td>0.04</td>
</tr>
</tbody>
</table>

4 OPTIMIZATION PROCESS

4.1 Response Surface Methodology (RSM)

Response surface methodology (RSM) is a collection of statistical and mathematical techniques useful for developing, improving, and optimizing processes. It also has important applications in the design, development, and formulation of new products, as well as in the improvement of existing product designs Figure 4.

The most extensive applications of RSM are in the industrial world, particularly in situations where several input variables potentially influence some performance measure or quality characteristic of the product or process. This performance measure or quality characteristic is called the response. It is typically measured on a continuous scale, although attribute responses, ranks, and sensory responses are not unusual. Most real-world applications of RSM will involve more than one response. The input variables are sometimes called independent variables, and they are subject to the control of the engineer or scientist, at least for purposes of a test or an experiment [6].

The origin of RSM is the seminal paper by Box and Wilson. They also describe the application of RSM to chemical processes. This paper had a profound impact on industrial applications of experimental design, and was the motivation of much of the research in the field. There have
also been four review papers published on RSM [7] [8] [9] [10]. The monograph by Myers [11] was the first book devoted exclusively to RSM.

In practice, complex process optimization problems such as this can often be solved by superimposing appropriate response surface contours. However, it is not unusual to encounter problems with more than two process variables and more complex response requirements to satisfy. In such problems, other optimization methods that are more effective than overlaying contour plots will be necessary.

4.2 Optimization Process

Normally, if you do not achieve the design goal that was repeated trial and error of the design change, you have to experiments and analysis again in conventional laboratory analysis method. It not only is inefficient but also spend the time and cost. The this problem was solved in one fell swoop statistical design by support of computing software, that the reason we use mathematical model to evaluation. We utilize the experimental design on computer, to determine the estimated equation characteristics evaluated, in order to optimize in mathematical programming shown on chart flow Figure 5. It can reduce to a minimum the number of experiments and analysis. The benefits of this method are: system was also applied to both linear and nonlinear phenomena, system design production efficiency is dramatically improved, it can be introduced economical system to all development and production sites, integrated system obtained a variety of design and analysis information and feel free to simple system in use.
4.3 Optimization Quarter-car Model

In this section we will focus to a single aspect of performance is related to the dynamics problem. We used the structure parameter which represented the following design variables: $k$, $c$, $b$, and $k_t$, with lower and upper boundaries show in Table 4. Simulation on mathematical quarter-car model was carried out to enable evaluation of vehicle performance measures. The resulted root-mean-squared (RMS) parameters, namely the $\text{RMS}(Z_s)$ sprung mass displacement and $\text{RMS}(Z_{t-def})$ tire deflection were taken into consideration under constraint function of $\text{RMS}(Z_{sus})$ suspension deflection limitation.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Lower boundary</th>
<th>Upper boundary</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>$k$</td>
<td>18000</td>
<td>30000</td>
<td>[Nm$^{-1}$]</td>
</tr>
<tr>
<td>$c$</td>
<td>600</td>
<td>1800</td>
<td>[Nsm$^{-1}$]</td>
</tr>
<tr>
<td>$b$</td>
<td>5</td>
<td>35</td>
<td>[kg]</td>
</tr>
<tr>
<td>$k_t$</td>
<td>50000</td>
<td>90000</td>
<td>[Nm$^{-1}$]</td>
</tr>
</tbody>
</table>

Optimal Both Displacement and Tire Deflection: Optimal solutions for mixed performance of sprung mass displacement and tire deflection: Optimal performance solutions for $\text{RMS}(Z_s)$ and $\text{RMS}(Z_{t-def})$ individually for parallel suspension network has been computed above.
Furthermore, it is also important to consider combined optimal vehicle performance across different measures. Here we present the results for a mixed $RMS(Z_s)$ and $RMS(Z_{t-def})$ measure:

$$RMS(\bar{Z}) = (1 - \alpha)RMS(Z_s) + \alpha RMS(Z_{t-def})$$

(12)

where $\alpha \in [0,1]$ is a weighting between $RMS(Z_s)$ and $RMS(Z_{t-def})$.

Equation 12 can be optimized with respect to the suspension parameters. The resulting optimal solutions are drawn for a particular $Z_s$ and $Z_{t-def}$. We use the same method SQP with RSM, we optimize modal parameters of the passive suspension system with inerter. The modal parameter results to optimal both displacement and tire deflection are presented in Table 5.

Table 5. The comparative modal parameters to optimal both displacement and tire deflection.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>k</th>
<th>c</th>
<th>b</th>
<th>k_t</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base model</td>
<td>24000</td>
<td>1200</td>
<td>0</td>
<td>70000</td>
</tr>
<tr>
<td>Optimization model</td>
<td>21818</td>
<td>1037</td>
<td>5</td>
<td>86363</td>
</tr>
</tbody>
</table>

Unit: [Nm$^{-1}$] [Nsm$^{-1}$] [kg] [Nm$^{-1}$]

In general, it can be seen that suspension with inerter offer performance advantages over conventional suspension for both $RMS(Z_s)$ and $RMS(Z_{t-def})$ combined. The Figure 6 shows that not only sprung mass displacement improves but also tire deflection reduces.

Although, both RMS displacement and tire deflection value are reduce slightly compare with individual optimization, near 8 percent in Table 6 and 12 percent of tire deflection improvement. All of that it shows the advanced results for vehicle oscillation when we apply inerter on suspension system.

Table 6. Comparison both RMS sprung mass displacement and tire deflection results.

<table>
<thead>
<tr>
<th>Model</th>
<th>RMS($Z_s$) [m]</th>
<th>Improvement</th>
<th>RMS($Z_{t-def}$) [m]</th>
<th>Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base model</td>
<td>0.369</td>
<td>0%</td>
<td>0.206</td>
<td>0%</td>
</tr>
<tr>
<td>Optimization model</td>
<td>0.339</td>
<td>7.84%</td>
<td>0.181</td>
<td>11.94%</td>
</tr>
</tbody>
</table>
5 VERIFICATION OPTIMAL MODAL PARAMETER

5.1 Verification Force Apply to Inerter

The Figure 7 shows that the force output at inerter component is the same shape and RMS value between mathematical and computational inerter mechanism with optimal value. At some points, the values are difference because of additional factors and coefficients inside computational model while the mathematical present with constant value in mass unit.

![Figure 7. Comparison RMS force between optimization of mathematical and computational inerter.](image)

5.2 Verification Sprung mass Displacement

For the post test analysis, we analyze output results of sprung mass displacement and tire deflection compare between mathematical and computational model in optimal parameters. Many ways of analyzing the results after tests have been performed, but we focus to the result of displacement and tire deflection in time domain which present for comfortable and handing on car.

Figure 8 shows that both mathematical and computational model have advantages compare with conventional model about sprung mass displacement. While the suspension without inerter has high displacement of sprung-mass, then the system applied inerter has a smaller one. The computational model has frictions inside, included mass components in inerter mechanism so the line curves are unlike with mathematical model at the same time.

![Figure 8. Comparison RMS sprung mass displacement among conventional, mathematical and computational model respectively.](image)
5.3 Verification Tire Deflection

From another point of view, the Figure 9 shows the comparison of tire deflection between mathematical and computational model, both of them have advantages compared to conventional model. While the suspension without inerter has high tire deflection then the systems applied inerter have better results. In this case, it still has a difference between computational model and mathematical model, because of the friction inside and coefficients of inerter mechanism.

![Figure 9. Comparison RMS tire deflection among conventional, mathematical and computational model respectively.](image)

6 CONCLUSIONS

We developed the verification process to numerically calculate the suspension model for quarter-car model, which is useful for estimate complex systems. It is noticed that all the computational models we have completed in the numerical models were based on a real model. The results shown that mathematical and computational models work on the same shapes of force apply to inerter and also had advanced on RMS displacement compare with conventional model. It is verified that mathematical model performance is correctly working theories, thus it is importance for optimization process.

The optimization process using response surface methodology which an optimal design to achieve variables stiffness, damping and inerter in suspension system with better results for both comfort and road holding. It was showed that conventional spring and damper always resulted in very normal vibration behavior, but the use of inerter can reduce the oscillation. Through system analysis, it was shown that the performance requirements can be achieved with optimization process. On the other hands, a verification process will be confirm system again, it shown the better results of system in small gap between mathematical and computational model. It is strongly verify that we optimization on mathematical model is correctly in this paper.

REFERENCES


Suspension Control Design for Vehicles. Published by Elsevier Ltd, 2010.


Vibration analysis of vehicle - UHPFRC Wild bridge interaction

Khanh Nguyen\textsuperscript{a}, Olga Rio\textsuperscript{a}, Marian Ralbovsky\textsuperscript{b}, Bernhard Freytag\textsuperscript{c}

\textsuperscript{a} Department of Construction  
Eduardo Torroja Institute for Construction Science  
Serrano Galvache No 4, 28033, Madrid, Spain  
khanh@ietcc.csic.es, rio@ietcc.csic.es

\textsuperscript{b} Mobility Department  
Austrian Institute of Technology  
Vienna, Austria  
marian.ralbovky@ait.ac.at

\textsuperscript{c} Laboratory for Structural Engineering  
Graz University of Technology  
Graz, Austria  
freytag@tugraz.at

\section*{ABSTRACT}

This paper presents an assessment of vibration perceived by pedestrians and by vehicle users in the slender UHPFRC-Wild bridge under heavy traffic load. A fully coupled vehicle-bridge interaction model is developed, in which the vehicle is modelled as a multibody system and interacts with the bridge’s finite element model by means of a contact implemented with the augmented Lagrange method. The road roughness surface is generated by considering the hypothesis of the isotropy and using the power spectral density proposed by [1]. An extensive study is performed to explore the influence of different factors like as road surface quality, vehicle velocity on the vibration-based human perception. Results show the importance of the road surface quality on the human comfort. The vibrations of bridge are not only governed by the fundamental mode of the bridge as defined by [2, 3], but also are dominated by the driving frequency and other torsional modes of the bridge, especially in presence of the road roughness. Moreover, an important reduction of ride comfort is observed when the road roughness quality decreases.

\section*{1 INTRODUCTION}

In order to establish the design criteria for vibration control in SLS there are two types of analysis procedure that can be performed: deflection- and acceleration-based methods. Using the deflection-based method, the vibrations are intended to be indirectly controlled by limiting the bridge deflection due to a static load. The American Association State Highway Transportation Officials (AASHTO) [4] specifies a deflection criteria of L/800 for vehicular bridges and L/1000 for bridges with footpath. It is perceived that these deflection limits are usually based on a rather arbitrary and sometimes conservative approach in designing bridges [5, 6]. Another deflection criteria employed in guidelines [7] is a pseudo-static approach based on Smith’s studies [8] in which the deflection limit is governed by the frequency of the first vertical mode of vibration. Debates on the necessity of deflection requirements in current bridge design specifications focus on the aspect that whether deflection limits provide effective control of bridge vibrations under normal truck traffic. Some works published in literature suggest that the deflection limits are not considered as the “good” method of controlling bridge vibrations [9, 6, 10].

Relation to the problem of vehicle-bridge interaction, a number of different approaches can be found in the literature, most of them attempting to improve the analytical model of the physical interaction between the bridge and the vehicles moving over it. The vehicle can be modelled with a simplest model as moving load [11, 12] or a more refined model of the moving load with Multi-Degrees-Of-Freedom (MDOF) to account for the dynamic properties of the vehicle [13, 14, 15, 16]. With the moving load model, the essential dynamic characteristics of the bridge can be captured with a sufficient degree of accuracy, however, it suffers from the drawback that the interaction between the bridge and the vehicle is not taken into account. While, the MDOF models allow consideration of the bouncing, rolling, yawing and pitching action of the moving vehicle.
relative to the bridge. Such effects are expected to be significant in the presence of road roughness or for vehicles moving at rather high speeds.

In this paper, the SLS of vibration of the UHPFRC-Wild bridge is assessed, focusing on the human comfort. A fully coupled vehicle-bridge interaction model is proposed: the bridge structures are modelled by means of Finite Element Model (FEM), the vehicle is considered as a multibody system and the interaction between vehicle-bridge through a contact implemented with the augmented Lagrange method. The results of an extensive number of nonlinear dynamic analyses are discussed in Section 4, considering the influence of different factors as road surface roughness quality, vehicle velocity, traffic. Finally, some conclusions are summarised in Section 5.

2 Wild Bridge Description and Modelling

2.1 Description

The Wild Bridge [17] is part of the new Eastern access of Völkermakt and crosses over a romantic valley. The principle of design of this bridge leaves the traditional way of solid construction and changes to sender and light structures made of UHPFRC. The arched structure is adopted due to the shape of the valley as shown in Figure 1. The rise of the arch is 18.30 m and the overall length of the bridge amounts to 157 m divided into: 9 spans of 15.0 m and 2 extreme spans of 11.0 m. Two very light and slender polygonal arches are arranged side by side with span of 69 m. Each arch consists of precast elements (six straight beams and eight node elements) which are reinforced by steel fibres but do not contain any conventional passive steel reinforcement. Eight node elements are situated at the bends of the arch. They are called as knee nodes. The beams are up to 16 m of length and have an orthogonal cross section measuring 1.2 m (see Figure 1). the walls are only 6 cm of thickness. The knee nodes have the same outer dimension as beams. A thickness of 20 cm is chosen for the walls, which leads to corresponding narrowing of the hollow space. The precast elements are assembled dryly and get a load carrying connection through the axial post-tensioning.

2.2 Finite Element Modelling

A three-dimensional linear elastic finite element model of the Wild Bridge was developed in Abaqus software [18]. Shell elements was adopted for the deck of bridge, taking into account the proper offsets of element plane to avoid the mass superposition at the intersections of the webs and slab. The arches and pillars were modelled by beam elements. Totally 7195 nodes, 6052 elements and 38370 degrees of freedom (DOF) were recognized in the FE model as shown in Figure 2.

In agreement with the information in the project, the mechanical properties of the FE model of the bridge are presented in the Table 1, including its designation, the adopted value and the respective unit.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Parameter</th>
<th>Unit</th>
<th>Adopted value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$E_a$</td>
<td>Elastic modulus of arches</td>
<td>GPa</td>
<td>48.0</td>
</tr>
<tr>
<td>$\rho_a$</td>
<td>Mass density of arches</td>
<td>kg/m$^3$</td>
<td>2500</td>
</tr>
<tr>
<td>$E_p$</td>
<td>Elastic modulus of bridge piers</td>
<td>GPa</td>
<td>38.0</td>
</tr>
<tr>
<td>$\rho_p$</td>
<td>Mass density of bridge piers</td>
<td>kg/m$^3$</td>
<td>2500</td>
</tr>
<tr>
<td>$E_d$</td>
<td>Elastic modulus of deck</td>
<td>GPa</td>
<td>35.0</td>
</tr>
<tr>
<td>$\rho_d$</td>
<td>Mass density of deck</td>
<td>kg/m$^3$</td>
<td>2500</td>
</tr>
<tr>
<td>$m_d$</td>
<td>Nonstructural mass on deck</td>
<td>kg/m$^2$</td>
<td>216.0</td>
</tr>
<tr>
<td>$h_d$</td>
<td>Thickness of deck</td>
<td>m</td>
<td>0.60</td>
</tr>
</tbody>
</table>
2.3 Modal analysis

A valuable understanding of the dynamic behavior of the bridges under the traffic loading can be anticipated by the simple inspection of the vibration modes. Figure 3 shows the first six modes of vibration of the considered bridge, which are used later in the model updating and the interpretation of the results presented in the following sections. Mode 1, 2 and 7 are associated with the transverse flexure, while mode 3, 4 and 5 are the vertical flexures.

3 Vehicle-Bridge Interaction Models

3.1 Vehicle model

Figure 4 shows a diagram of the vehicle model used in the dynamic analysis of vehicle-bridge interaction. This model is similar to that employed by several authors [13, 14, 15, 16] with the mechanical properties corresponding to the H20–44 truck model defined by the American Association of State Highway and Transportation Officials (AASHTO) specifications [4]. The vehicle is mod-
Figure 3. The first six modes of vibration of Wild Bridge obtained by numerical model

elled as a multibody system composed by individual rigid bodies (vehicle body and two axles). The vehicle body and the axles are connected by the suspension system, which is modelled by the linear spring-dashpot elements. The types are considered as the linear spring-dashpot elements, in which the bottom node has a contact with the bridge surface.

Figure 4. Vehicle model in dynamic analysis: (a) side view, (b) front view

The vehicle body is assigned five DOFs: vertical displacement $z_c$, pitching motion $\theta^c$, rolling motion $\theta^c_x$, with associated mass $m_c$ and mass moments of inertia $J^c_z$, $J^c_x$. Each axle has two DOFs: vertical displacement $z_{a,i}$, rolling motion $\theta^a_{x,i}$ and the corresponding mass $m_{a,i}$ and mass moment of inertia $J^a_{x,i}$ ($i = 1, 2$ is the index for the front and rear axle, respectively). In total, the vehicle has 7 DOFs. The equation of motion of the vehicle can be expressed as
\[ M_v \ddot{x}_v + C_v \dot{x}_v + K_v x_v = F_v \]  

where \( F_v \) is the external force vector applied on the vehicle; \( M_v, \ C_v, \ K_v \) are the mass, damping and stiffness matrices of the vehicle, respectively (see Appendix A for more detail). In general, the multibody dynamics equilibrium equation may include second order and nonlinear terms related to the inertial forces, but in this work these terms may be linearized or neglected.

### 3.2 Interaction

Vehicle-bridge interaction is established by means of a contact between the bottom node of the tyre elements and the bridge surface. The contact process is divided into two phases: initially, the identification of surfaces in contact is performed and then the necessary kinematic relations to enforce the corresponding contact constraints are set up. In the current research, the contact formulation “node to surface” is used in the first phase and the augmented Lagrange method is applied for the second phase.

The proposed methodology is developed within Abaqus software [18], which provides the capabilities of modelling the bridge structure by finite elements and the vehicle by multibody system. The vehicle, the bridge and the interaction between both form a nonlinear coupled system that the global system of equations can be expressed by the following matrix form, including the interaction force:

\[
\begin{bmatrix}
M_v & 0 \\
0 & M_b
\end{bmatrix}
\begin{bmatrix}
\ddot{x}_v \\
\dot{x}_b
\end{bmatrix}
+
\begin{bmatrix}
C_v & 0 \\
0 & C_b
\end{bmatrix}
\begin{bmatrix}
\dot{x}_v \\
\dot{x}_b
\end{bmatrix}
+
\begin{bmatrix}
K_v & 0 \\
0 & K_b
\end{bmatrix}
\begin{bmatrix}
x_v \\
x_b
\end{bmatrix}
=
\begin{bmatrix}
F_v \\
F_b
\end{bmatrix}
\]  

where \( F_b \) is the external force vector applied on the bridge; \( M_b, \ C_b, \ K_b \) are the mass, damping and stiffness matrix of the bridge, respectively; \( F_v^C \) is force vector applied on the vehicle as consequence of the interaction with structure, and \( F_b^C \) their counterparts on the structure.

In order to solve the system of differential equation (2) in the time domain, the HHT-\( \alpha \) implicit integration method [19] is used. A constant time step of 0.001 s is adopted, which is small enough to accurately capture high frequency vibrations and to allow for the precise definition of the roughness profile.

Furthermore, to evaluate the ride comfort of the road vehicle by using the criteria specified in ISO 2631-1 [20], the time histories of the vertical acceleration responses of the vehicle at driver seat \( \ddot{z}_s \) should be determined. The time histories of vertical and rotational acceleration at the centroid of the car-body can be obtained directly. Since the vehicle is considered as rigid body, the vertical acceleration can be computed by the following expression

\[
\ddot{z}_s = \ddot{z}_c + l_x \dot{\theta}_c^r - b_x \dot{\theta}_x^r
\]  

where \( l_x, b_x \) are relative distances between the centroid of car-body and the driver seat (see Figure 4).

### 4 Numerical assessment of vehicle-induced bridge vibrations

In accordance with the design of the Wild bridge, the deck has been designed to support two road lanes (3.5 m wide each) and has only one sidewalk as shown in Figure 1(b). With this design, the road axis is eccentric 0.35 m respect to the bridge axis (see Figure 5(a)), which implies that the vehicles run over the bridge with certain eccentricity. Therefore, two load cases have been considered in the study of influence of road surface quality and of vehicle velocity on the dynamic bridge responses: (i) Load Case I with an eccentric vehicle in the lane 2 \( (e = 2.1 \text{ m}) \); Load Case II with an eccentric vehicle in the lane 1 \( (e = 1.4 \text{ m}) \), as shown in Figure 5. Although the cross slope is 2% for the road, it is not included in this study, due to that the influence of the cross slope on the bridge acceleration is weak [16]. External platforms have been considered at both abutments.
in all calculations in order to study the possible effect generated during the time that the vehicle moves from the external platform to entering onto the deck and leaving the bridge. The platforms are connected to the ground by vertical springs that simulate the pavement and soil flexibility.

The FEM of Wild bridge developed in Section 2 was used in the dynamic analyses in order to assess the vehicle-induced bridge vibrations. An extensive number of analyses with different factors related to the vehicle action is discussed below.

4.1 Influence of road surface quality

One of the most important sources of dynamic excitation in the vehicle-bridge interaction is the road roughness. Generally, it can be given through an ergodic stationary Gaussian random process, described by its power spectral density (PSD). In this study, the one-sided PSD proposed by ISO 8608:1995 [1] is employed:

\[ G(n) = a \left( \frac{n}{0.1} \right)^{-2} \]  

being \( a \) the spectral roughness coefficient \([\text{m}^3/\text{cycle}]\) whose value is chosen depending on the road surface quality, \( n \) the spatial frequency \([\text{cycle/m}]\). According to ISO 8608:1995 the following road qualities and keyword to refer the results are considered in this work: very good (road A) \( a = 16 \times 10^{-6} \); good (road B) \( a = 64 \times 10^{-6} \); regular (road C) \( a = 256 \times 10^{-6} \), bad (road D) \( a = 1024 \times 10^{-6} \).

In order to generate the appropriate road roughness profile under the left and right wheels so that there is a good coherency between left and right profiles, the hypothesis of the isotropy of the road surface is adopted here. According to [21, 22] the autocorrelation \( (R_L(\delta), R_R(\delta)) \) and cross-correlation \( (R_{LR}(\delta), R_{RL}(\delta)) \) functions of two profiles must satisfy the following conditions for the homogeneity:

\[ R_L(\delta) = R_R(\delta) = R(\delta) \]  
\[ R_{LR}(\delta) = R_{RL}(\delta) = R_x(\delta) \]

Moreover, because of isotropy the relation between the cross-correlation and autocorrelation functions can be obtained as:

\[ R_x(\delta) = R(\sqrt{\delta^2 + 4b^2}) \]

where \( \delta \) is the distance between two points in longitudinal direction and \( 2b \) is the distance of two parallel profiles. The one-sided direct and cross spectral densities of the profiles can be defined by Fourier transformation:
Thus from a knowledge of the one-sided PSD $G(n)$, the autocorrelation function $R(\delta)$ can be obtained by the inverse Fourier transformation of the one-sided PSD. The cross-correlation function $R_x(\delta)$ and the one-side cross PSD $G_x(n)$ are later determined using the equation (7) and (9), respectively. The road roughness profile can be generated as the sum of a series of harmonics:

$$r_1(x) = \sum_{i=1}^{N} \sqrt{2G(n_i)\Delta n} \cos(2\pi n_i x + \phi_i)$$

and the second parallel profile at distance $2b$ is defined by [23]:

$$r_2(x) = \sum_{i=1}^{N} (\sqrt{2G_x(n_i)\Delta n} \cos(2\pi n_i x + \phi_i) + \sqrt{2(G(n_i) - G_x(n_i))\Delta n} \cos(2\pi n_i x + \theta_i))$$

in which $N$ is the number of discrete frequencies $n_i$ in range $[n_{\text{min}}, n_{\text{max}}]$, $\Delta n$ is the increment between successive frequencies, $\phi_i$ is the random phase angle uniformly distributed from 0 to $2\pi$, $\theta_i$ is other random uniformly distributed phase angles.

For the dynamic analysis, in order to achieve some statistical significance, ten sets of left and right roughness profiles are generated for each road surface quality. The range of frequency of interest from 0.01 to 10 cycle/m as recommended by ISO 8608:1995 [1] was considered in this work.

To illustrate the influence of road roughness quality, firstly, the deck acceleration responses obtained in the time domain are interpreted in the frequency domain in order to analyse the contribution of different frequency sources on the response, using its Fourier transformation. Figure 6 shows the frequency content of deck acceleration at point C1 (see Figure 5(b)) when the vehicle crosses over the bridge at $v = 120$ km/h with and without considering the road roughness. The following observations are extracted from Figure 6:

1. The driving frequency ($f = v/2L = 1.11$ Hz for $v = 120$ km/h) and the frequency of the first vertical mode have relevant contribution to the vertical deck acceleration. Furthermore, if the passing vehicle is more eccentric with respect to the bridge axis (Load Case I), the torsional modes have more effect.

2. As the passing vehicle is closer to the monitored point, the influence of driving frequency on the response at this point increases.

3. The presence of road roughness amplifies significantly the contribution of higher modes on the dynamic response of deck, specially in range $[5-75]$ Hz.

Figure 7(a) shows the average peak vertical acceleration $\mu$ along sidewalk and along the cross section at the point C1 when the vehicle runs over the bridge at $v = 120$ km/h and the Load Case I is considered, considering different road qualities as well as a perfect road without irregularities. The dispersion of the results obtained from ten sets of profiles is represented as a coloured band centred on the average value for each road quality. This band width corresponds to the minimum and maximum value at each point. It can be observed that the larger the spectral roughness coefficient
Figure 6. Frequency content of deck acceleration response at point C1 when the vehicle runs over the bridge at $v = 120$ km/h with and without road roughness: (a) Load Case I, (b) Load Case II.

$a$, i.e. the worse the road quality, the larger the effect of road roughness on the dynamic response. In particular, if the road quality is regular (road C), which can represent the status of “minor” or “secondary” roads the maximum vertical acceleration obtained at the point $C_1$ can be up to 8 times higher than the case with perfect road, and the maximum acceleration allowed by EN 1992-2: Eurocode 2 [3] is obviously exceeded along the sidewalk (see Figure 7(a)). This clearly explains the importance of the road quality from the point of view of the pedestrian’s comfort and of the fatigue. Moreover, from Figure 7(a) an important effect can be observed that is the impact effect when the vehicle enters and leaves the bridge. This effect is generated due to the differential vertical stiffness between the external platform and the deck and is the source of the high vertical acceleration at the deck, nearby the abutment. Such peak acceleration would far exceed the admissible limit in the SLS of vibration ($a_{lim}$) and would be hardly affected by the road roughness.

The reactions at various magnitudes depend on passenger expectations with regard to trip duration and the type of activities passengers expect to accomplish and many other factors (acoustic noise, temperature, etc.). Therefore, the acceptable values of vibration magnitude for comfort can not be well defined, but some values can give approximate indications of likely reactions to various magnitudes in public transport as defined in ISO 2631 [20]. From range lists defined in ISO 2631 [20], the acceleration $a_u$ of 1 m/s$^2$ is adopted to distinguish uncomfortable vibrations for this study. Ride comfort factor is used here to interpret the results. This factor is computed as the coefficient between the uncomfortable threshold $a_u$ and the maximum vertical acceleration at driver seat $a_{max}$. Figure 7(b) displays the ride comfort factor calculated for different road roughness quality. It can be seen that the road roughness quality is also very important for people inside the car, or in other words, the poorer road roughness quality, the more uncomfortable the people receive.

4.2 Influence of vehicle velocity

A range of vehicle velocities that corresponds to the possible speeds of the truck over the bridge, ranging from 60 to 120 km/h, in increments of 10 km/h, has been considered. Figure 8(a) shows the peak vertical acceleration recorded at the point $C_1$ in the sidewalk for different vehicle velocities. Due to the fact that the energy input to the bridge is higher when the vehicle moves over the bridge at a higher velocity, the amplitude of the acceleration responses of the bridge increases as
the vehicle velocity increases [24]. This trend is clearly observed in Figure 8(a) for the case that the road surface is perfect, i.e. without roughness. However, with the presence of even small road roughness (road A), the amplitude of acceleration responses of the bridge does not always increase with increasing the vehicle speed. As was mentioned in the previous section, the road roughness amplifies significantly the contribution of higher frequency on the dynamic responses of the deck, specially in range $[5 − 75] \text{ Hz}$, such a fact is reconfirmed by the frequency content of vertical acceleration at the point $C_1$ for different vehicle velocities plotted in Figure 8(b). Clearly, this effect is greater for $v = 60 \text{ km/h}$ than for $v = 90 \text{ km/h}$ and for $v = 120 \text{ km/h}$. This explains why the peak acceleration obtained with $v = 60 \text{ km/h}$ is greater than the others. Furthermore, the influence of the driving frequency is greater at a higher vehicle velocity.

5 CONCLUSIONS

In this paper, the assessment of serviceability limit state of vibrations in the UHPFRC-Wild bridge under traffic loading is carried out by means of FEM of Wild bridge and nonlinear dynamic analysis of vehicle-bridge interaction. Based on the analysis results, the following conclusions could be made:

- Road surface roughness is important factor which greatly affects the SLS of vibrations of the Wild-bridge. With the deterioration of the road surface quality, the vibrations of the Wild-bridge increases distinctly and can easily exceed the admissible level for pedestrians, which is necessary to maintain an adequate road surface quality.

- The vertical vibrations of bridge not only depend on its first vertical frequency but also depend on other frequencies related to torsional modes, especially when the vehicle moves over the bridge with a certain eccentricity respect to bridge axis and the road roughness is presented. Therefore, the first vertical frequency of the bridge, as proposed by some codes and guidelines [3, 7], may not effectively control bridge vibrations.

- The impact effect generated during the vehicle entering and leaving the bridge is very important in terms of the peak acceleration of the deck recorded close to the abutments. Due

---

**Figure 7.** Influence of road roughness quality: (a) peak vertical acceleration along sidewalk (the one passes through the point $C_1$), (b) peak vertical acceleration for different positions across the deck width at the point $C_1$. Load Case I and vehicle velocity $v = 120 \text{ km/h}$
to this effect, the peak acceleration is very high and clearly exceeded the admissible level. Although the duration of this impulse is very short, no code or guideline (known by authors) takes it into account. Further investigation is required in this regard.

- The vehicle velocity does not have a clear influence on the Wild-bridge vibrations. With the presence of even small road roughness, the amplitude of acceleration responses does not always increase with increasing the vehicle speed.

Acknowledgements
The authors thank to other members of the team of Intitut für Betonbau at Technical University of Graz: Lutz Sparowitz, Michael Reichel, who provided the necessary information of the Wild bridge design. K. Nguyen and O. Rio also thank to the MINECO of Spain for supporting of project BIA2013-48480-C2-1R.

REFERENCES


A Vehicle characteristics and its matrices

The main characteristics of the selected vehicle are \( l_1 = 3.153 \text{ m}, l_2 = 1.577 \text{ m}, a_1 = 1.41 \text{ m}, a_2 = 2.05 \text{ m}, m_c = 17000 \text{ kg}, m_{a,1} = 600 \text{ kg}, m_{a,2} = 1000 \text{ kg}, J_{y}^{c} = 90 \times 10^{3} \text{ kg.m}^2, J_{x}^{c} = 13000 \text{ kg.m}^2, J_{z,1} = 550 \text{ kg.m}^2, J_{z,2} = 600 \text{ kg.m}^2, k_{s,3} = k_{s,4} = k_{s,1} = 16 \times 10^{3} \text{ N/m}, k_{s,1} = k_{s,2} = k_{s,5} = 373 \times 10^{3} \text{ N/m}, k_{r,1} = k_{r,2} = k_{r} = 1.57 \times 10^{6} \text{ N/m}, k_{f,3} = k_{f,4} = k_{f} = 785 \times 10^{3} \text{ N/m}, c_{r,1} = c_{r,2} = c_{r} = 35 \times 10^{3} \text{ N s/m}, c_{s,3} = c_{s,4} = c_{s,1} = 25 \times 10^{3} \text{ N s/m}, c_{f,1} = c_{f,2} = c_{f} = 2 \times 10^{2} \text{ N s/m}, c_{r,3} = c_{r,4} = c_{f} = 1 \times 10^{2} \text{ N s/m}.

The displacement vector and mass matrix are

\[
\dot{x}_v = \{z_c, \theta_y^c, \theta_y^c, \theta_{z,1}, \theta_{z,1}, z_{a,2}, \theta_{z,2}, \theta_{z,2}\},
\]

\[
M_v = \text{diag}\{m_c, J_x^c, m_{a,1}, J_{z,1}^a, m_{a,2}, J_{z,2}^a\}.
\]

The stiffness and damping matrices of vehicle have the same structure. For instance, the stiffness \( K_v \) is

\[
K_v = \begin{bmatrix}
k_{11} & k_{12} & 0 & k_{14} & 0 & k_{16} & 0 \\
k_{12} & k_{22} & 0 & k_{24} & 0 & k_{26} & 0 \\
k_{33} & 0 & k_{35} & 0 & k_{37} & 0 & 0 \\
0 & 0 & 0 & k_{44} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & k_{55} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & k_{66} & 0 \\
\text{sym.} & & & & & & k_{77} \\
\end{bmatrix}
\]

where:

\begin{align*}
k_{11} &= 2(k_{s,1} + k_{s,2})/2; \\
k_{12} &= 2k_{s,1}l_2 - k_{s,2}/l_1; \\
k_{14} &= 2k_{s,1}; \\
k_{16} &= 2k_{s,1}; \\
k_{22} &= 2k_{s,1}l_2; \\
k_{24} &= -2k_{s,1}l_2; \\
k_{26} &= 2k_{s,1}/l_2; \\
k_{33} &= \frac{a_1^2}{2}(k_{s,1} + k_{s,2})/2; \\
k_{35} &= \frac{a_1^2}{2}k_{s,1}; \\
k_{37} &= \frac{a_1^2}{2}k_{s,2}; \\
k_{44} &= 2(k_r + k_{s,1})/2; \\
k_{55} &= (a_1^2k_r - a_2^2k_{s,1})/2; \\
k_{66} &= 2(k_f + k_{s,2})/2; \\
k_{77} &= (a_1^2k_f - a_2^2k_{s,2})/2
\end{align*}
Vibrations of a rolling tyre

Ivan F. Kozhevnikov *

* Institution of Russian Academy of Sciences
Dorodnicyn Computing Centre of RAS (CC RAS)
Vavilov st. 40, 119333, Moscow, Russia
kogevnik@ccas.ru

ABSTRACT
We investigate vibrations of an unloaded and loaded tyre rolling at constant speed without slipping in the contact area. A previously proposed analytical model of a reinforced tyre is considered. The surface of the tyre is represented by flexible tread, combined with parts of two tori (sidewalls of the tyre). The elastic sidewalls are described by the Mooney-Rivlin model of incompressible rubber. The tread is reinforced with inextensible cords. In the undeformed state, the tread is represented by a circular cylinder. The contact between the wheel and the ground plane occurs by the part of the tread. The natural frequencies (NF) are determined numerically and mode shapes (MS) are determined analytically for loaded rotating tyre. The results were compared with experiments for the non-rotating tyre. In the case of loaded rotating tyre, the increasing of the angular velocity of rotation implies that NF decrease. Moreover, a phenomenon of frequency loci veering is visible here: NF as functions of angular velocity approach each other and then veer away instead of crossing. The MS interact in veering region and, as a result, interchange.

Keywords: Radial tyre, Analytical model, Rolling, Modal analysis, Vibrations.

1 INTRODUCTION
The vibrations of tyres were studied by many authors. The vibration transmission properties of the tyre between the tyre-road contact point and the wheel axle are analysed using a flexible ring model [1]. Precession of the standing waves in a thin elastic ring rotating with constant angular velocity, was first observed in [2]. It was shown that if the standing waves were excited in the ring and if the ring rotates with a constant angular velocity, then the wave turns on a different angle with respect to inertial space depending on the number of MS. The vibrations of flexible extensible rotating ring are considered in [3] taking into account the geometrical non-linearity. Paper [4] presents a survey of wheel vibrations in a complex dynamic vehicle suspension system and their influence on the forces transduced in a high frequency area from the tyre to the vehicle’s body. Secondly, it also presents the transient evolution of tyre models used for prediction and understanding high frequency movements in the tyre contact area, producing the guiding forces and torques during vehicle handling. The effects of rotation on the NF of a loaded tyre have been studied in [5] using FE model. In particular, the phenomenon of a frequency loci veering is considered. This phenomenon consists of mutually repulsive behaviour of the NF, and is induced by the aperiodicity of the tyre resulting from its deformations under the load. A model of a reinforced tyre was proposed in [6]. In the case of wheel rolling without slipping in an unknown in advance contact area, the complete system of equations of motion was obtained. The steady-state regime of rolling at constant speed was investigated. In this paper, we study the vibrations of an unloaded and loaded tyre rolling at constant speed. This tyre model was also used in studying of the vibrations of an unloaded and loaded non-rotating tyre [7].

2 MODELLING OF A WHEEL WITH A REINFORCED TYRE
Assume that the wheel with a reinforced tyre consists of disc (0) joined to the sidewalls (1,2) and of tread (3) (Figure 1 (a)). The wheel disc is a rigid body with six degrees of freedom. In the
undeformed state, the sidewalls are represented by parts of two tori. The elastic sidewalls are described by the Mooney-Rivlin model [8] of incompressible rubber. The tread is reinforced with inextensible cords. In the undeformed state, the tread is represented by a circular cylinder of radius $r$ and height $2l$ (tread width).

Introduce an inertial frame $OX_1X_2X_3$ (IF), such that the tyre contacts by its tread with the ground plane $OX_1X_2$ and a moving frame $C_{x_1}x_2x_3$ (MF) with its origin $C$ coincident to the mass centre of the disc (Figure 1 (a)). Denote $(X_1, X_2, X_3)$ the coordinates of point $C$ in the IF. The position of the tread is determined by the rotation angles $\beta_0, \theta$ about the axes $OX_3$ and $Cx_2$ respectively. We define the surface of tread for $\beta_0$ and $\theta$ fixed in the form:

$$r_3(\phi, \xi, t) = \sum_{i=1}^{3} X_i l_i + \Gamma_3(\beta_0) \Gamma_2(\vartheta) \left( r e_1 + l_\xi e_2 + r \sum_{i=1}^{3} U_i(\phi, \xi, t) e_i \right), \quad \varphi \mod 2\pi \xi \in [-1; 1]$$

Here $l_i, e_i$ are unit vectors of axes $OX_i$ and $Cx_i$ respectively; the value of parameter $\xi = 1 (\xi = -1)$ corresponds to the linking line of the tread and the first (second) sidewall, the value $\xi = 0$ corresponds to median line $l_0$ of the tread; $rU_i(\phi, \xi, t)$ are components of displacement vector of points of tread in the MF. The tread deformations are considered taking into account the exact nonlinear conditions of inextensibility of reinforcing cords [6]. Due to nonlinear geometric constraints in the deformed state, the tread retains its cylindrical shape, which is not circular for a typical configuration. The condition of the inextensibility of $l_0$ reads

$$2(u + v') + (u + v')^2 + (v - u')^2 = 0$$

Here $ru(\phi, t)$ and $rv(\phi, t)$ are, respectively, the radial and tangential components of the displacement vector

$$ru(\phi, t)e_1 + rv(\phi, t)e_2 - rv(\phi, t)e_3$$

Figure 1. The model of a wheel with a reinforced tyre.
of points of \( l_0 \) in the MF (Figure 1 (b)).

It was assumed [6] that the contact area of the tyre and the plane \( OX_1X_2 \) can be represented by a rectangle of constant width \( 2l \), equals to the tread width, and of variable length \( r(\varphi_2(t) - \varphi_1(t)) \). The length is defined by two functions of time \( \varphi_1(t), \varphi_2(t) \), which are unknown in advance (Figure 1 (b)). These functions can be obtained from the equations of motion. The values \( \varphi \in L_1 = [\varphi_1(t), \varphi_2(t)] \) correspond to contact area, and the values \( \varphi \in L_2 = [\varphi_2(t), 2\pi + \varphi_1(t)] \) correspond to the free surface of the tyre.

In the contact area \( L_1 \) the holonomic constraint is given by

\[
\mathbf{r}_3 = -r \left( \varphi - \frac{\pi}{2} \right) \mathbf{l}_1 + l \mathbf{\xi}_2, \quad \varphi \in L_1
\]  

(3)

In this relation, without loss of generality, we have assumed that the rolling of the wheel occurs along the axis \( OX_1 \), when the median line \( l_0 \) of the tread coincides with this axis (the angle \( \beta_0 = 0 \)).

The conditions (1) and (3) allow the determination of displacements of points of \( l_0 \) in the contact area. We also assume that \( X_2 = 0 \) and therefore \( w = 0 \), i.e. the mass centre of the disc does not move laterally. Suppose that the wheel rolls without slipping and without jumping. This means that the velocity of points of \( l_0 \) in the contact area \( L_1 \) are equal to zero.

The equations of motion and conditions on the unknown in advance boundary of contact area were obtained [6] from the Hamilton-Ostrogradsky variational principle for nonconservative systems

\[
\int_{t_1}^{t_2} (\delta T + \delta A) dt = 0
\]

The kinetic energy of the wheel \( T \) consists of kinetic energy of the disc and the kinetic energy of the tyre assuming that the whole mass of the tyre is distributed uniformly along \( l_0 \) with linear density \( \rho \). The work \( \delta A \) at virtual displacements has the following structure

\[
\delta A = \delta A_F + \delta A_P + \delta N_1 + \delta N_3 + \delta N_6
\]

Here \( \delta A_F \) is the work performed by the external forces, the work \( \delta A_P \) is the work performed by the potential forces (it comprises the work performed by the pressure and the variation of potential energy of the rubber stretching in the Mooney-Rivlin model when the sidewalls and the tread are deformed), the works \( \delta N_1, \delta N_3, \delta N_6 \) are performed by the reactions of the constraints.

### 3 Unloaded Tyre

Suppose that the unloaded wheel rotates with constant angular velocity \( \Omega \). Then

\[
X_1 = \text{const}, \quad X_3 = \text{const}, \quad \dot{\theta} = \Omega
\]

The equations of motion and the condition of inextensibility of \( l_0 \) (2) read

\[
-\rho r^3 \left( \ddot{u} - \Omega^2 (1 + u) - 2\Omega \dot{v} \right) + n_{11} u'' - n_{01} u + (m_{20} - m_{21}) v' - n_0 + \lambda (1 + u + v') - \left[ \lambda (u' - v) \right]' = 0
\]

\[
-\rho r^3 \left( \ddot{v} - \Omega^2 v + 2\Omega \dot{u} \right) + (m_{02} - m_{12}) u' + n_{12} v'' - n_{02} v - \left[ \lambda (1 + u + v') \right]' - \lambda (u' - v) = 0
\]

\[
2(u + v') + (u + v')^2 + (v - u')^2 = 0 \quad \Rightarrow \quad u \cong -v'
\]

(4)

where the Lagrange multiplier \( \lambda = \lambda (\varphi, t) \) determines the tension of \( l_0 \).

**Remark 1.** The constant coefficients \( n_0, n_{01}, n_{11}, n_{02}, n_{12}, m_{21}, m_{12}, m_{20}, m_{02} \) are determined analytically in elementary functions by evaluating definite integrals (by integrating over sidewalls and over tread of the tyre) and depend on the geometric parameters of the tyre and on the internal tyre pressure \( p \). Usually, in the phenomenological approach the coefficients are unknown and are found from experiment.
In our work, analytical expressions for the coefficients of the linearised condition of the inextensibility of the rotating tyre (due to the centrifugal forces of inertia added to the pressure inside the tyre) are to be retained in (6), i.e.

\[ p = \rho r^3 \Omega^2 \]

Here \( \lambda_1 \) is of first order of smallness. In this case, the tension of rotating tread increases (compared with non-rotating tyre) due to the centrifugal forces of inertia added to the pressure inside the tyre. Differentiating both sides of the first equation of (4) and adding it to the second equation, using the linearised condition of the inextensibility of \( I_0 \), we obtain the equation for the function \( v(\varphi, t) \) and the condition of its periodicity

\[
\rho r^3 v'' + 4\rho r^3 \Omega v' - \rho r^3 \varphi + (a_0 - g_0) v^{(4)} + (a_1 - 3g_0) v'' + a_2 v = 0, \quad v(0) = v(2\pi)
\]

The natural vibrations of the tyre, described by the equation (5), can be obtained in the form

\[ v = e^{i\omega t} X(\varphi), \quad u = -v' = -e^{i\omega t} X'(\varphi) \]

where \( \omega \) is an angular frequency. Then

\[
(a_0 - g_0)X^{(4)} + (a_1 - 3g_0 - \rho r^3 \varphi^2)X'' + 4\rho r^3 \Omega \omega i X' + (a_2 + \rho r^3 \omega^2) X = 0
\]

The solution of this equation can be represented in the form

\[ X(\varphi) = G_1 e^{p_1 \varphi} + G_2 e^{p_2 \varphi} + G_3 e^{n \varphi} + G_4 e^{n \varphi} \]

where \( p_j, j = 1, ..., 4 \) are the roots of the characteristic equation

\[
(a_0 - g_0)p^4 + (a_1 - 3g_0 - \rho r^3 \omega^2)p^2 + 4\rho r^3 \Omega \omega i p + (a_2 + \rho r^3 \omega^2) = 0
\]

Since the function \( X(\varphi) \) must be 2\( \pi \)-periodic, only exponents with pure imaginary indices need be retained in (6), i.e. \( p = in, n \in Z \). The characteristic equation (7) reads

\[
A(n^2)\omega^2 + B(n, \Omega)\omega + C(n^2, \Omega^2) = 0
\]

\[ A(n^2) = \rho r^3 (1 + n^2), \quad B(n, \Omega) = -4\rho r^3 n\Omega, \quad C(n^2, \Omega^2) = (a_0 - g_0)n^4 - (a_1 - 3g_0)n^2 + a_2 \]

The infinite frequency spectrum of an unloaded rotating tyre can be found analytically from the characteristic equation (8). Thus, the NF of an unloaded rotating tyre \( \omega_n \) are expressed in terms of NF of an unloaded non-rotating tyre \( \omega_n^0 \)

\[ \omega_n = \frac{2n}{1 + n^2} \Omega \pm \sqrt{(\omega_n^0)^2 + \frac{n^2(3n^2 - 1)^2}{(n^2 + 1)^2} \Omega^2}, \quad \omega_n^0 = \pm \sqrt{-\frac{-a_0n^4 + a_1n^2 - a_2}{\rho r^3 (1 + n^2)}}, \quad n \in Z \]

In our work, analytical expressions for the coefficients \( a_0, a_1, a_2 \), determining the NF of an unloaded non-rotating tyre, have a fairly complicated form (the coefficients depend on the geometric parameters of the tyre and on the internal tyre pressure) and are calculated in terms of elementary functions.

The plot of NF as a function of angular velocity is shown in Figure 2 (a) for Input I (see A). The experimental NF [9], corresponding to an unloaded non-rotating tyre, are plotted as black squares. On the y-axis the two branches grow from each point corresponding to the NF of an unloaded non-rotating tyre (the top branch corresponds to \( \omega_n > 0 \), the bottom branch corresponds to \( \omega_{-n} > 0 \),
Figure 2. The NF of an unloaded rotating tyre (Input I) as a function of angular velocity, (a) Lagrangian specification, (b) Eulerian specification. The black squares ■ are the experimental NF.

where $n > 0$), i.e. each NF of an unloaded non-rotating tyre corresponds to two NF of an unloaded rotating tyre. Note that if $n = 1$, then the top branch corresponds to $\omega_1 = \Omega + \omega_1^0$, and the bottom branch corresponds to $\omega_{-1} = -\Omega + \omega_1^0$. In this case, $\omega_{-1}$ vanishes at $\Omega = \omega_1^0 = 2\pi v_1 = 637.74$ rad·s$^{-1}$. 


Making the change of variables $\varphi$ by $\alpha = \varphi + \Omega t - \pi/2$ we pass from the Lagrangian specification to the Eulerian specification. In this case, one should use the equation (11) as we shall see below instead of the characteristic equation (7). An infinite spectrum of NF can be represented in the form

$$ \omega_n = \frac{-n(n^2-1)}{1+n^2} \Omega + \sqrt{\frac{(\omega_0^2)^2 + n^2(n^2-1)^2}{(n^2+1)^2}} \Omega^2, \quad n \in \mathbb{Z} $$

The plot of NF as a function of angular velocity is shown in Figure 2 (b). The top branch corresponds to $\omega_{-n} > 0$, the bottom branch corresponds to $\omega_n > 0$, where $n > 0$. Note that if $n = 1$, then $\omega_2 = (\omega_0^2)^2$ does not depend on the angular velocity of rotation.

4 LOADED TYRE

Consider the problem of vibrations of a tyre about the steady-state regime of rolling at constant speed without slipping in the contact area. The steady-state regime of rolling of a loaded tyre was considered in [6]. Suppose that

$$ X_1 = r\Omega, \quad X_3 = \text{const}, \quad \dot{\theta} = \Omega $$

Making the change of variables $\varphi$ by $\alpha = \varphi + \Omega t - \pi/2$ we pass from the Lagrangian specification to the Eulerian specification. We represent the functions determining the shape of the deformed tread, determining the contact area and the Lagrange multipliers in the form

$$ u(\varphi, t) = U(\alpha) + U_{\varphi}(\alpha, t), \quad v(\varphi, t) = V(\alpha) + V_{\varphi}(\alpha, t), \quad \lambda(\varphi, t) = \lambda^o(\alpha) + \lambda_{\varphi}(\alpha, t) $$

The Lagrange multipliers $\mu_{11}, \mu_{12}$ describes the longitudinal component of the reaction of the constraint at the boundary points ($\varphi = \varphi_1, \varphi = \varphi_2$) of the contact area. The boundaries of contact area are defined by two functions $\alpha_1(t), \alpha_2(t)$ (Figure 1 (b)), which are unknown in advance. We take into account the differentiation rules

$$ \dot{u}(\varphi, t) = U'(\alpha)\dot{\alpha} + U_{\varphi}(\alpha, t)\dot{t} + U_{\varphi}(\alpha, t) $$

Then, the solution describing the steady-state motion $U(\alpha), V(\alpha), \lambda^o(\alpha), \mu_{1k}, \alpha_{k}'$ and satisfying the equations and boundary conditions can be found [6] from the system

$$ g_0(1 + U - U'' + 2V') - n_0 - n_01U + n_11U'' - (m_21 - m_20)V' + \lambda^o - \lambda_{\varphi} = 0 $$

$$ 2(U + V') + (U + V')^2 + (V - U')^2 = 0 \implies U = -V' $$

Here $\lambda^o = n_0 - g_0 + \lambda_1^o$, where $\lambda_1^o$ is of first order of smallness, $|h(\vartheta)|_{\varphi} = h(\vartheta_1 + 0) - h(\vartheta_2 - 0)$ is the jump in the function at the final point of the contact area. The index $l(k)$ means that:

$$ |h(\vartheta)|_{l(k)} = h(\vartheta_1 + 0), \quad |h(\vartheta)|_{l(k)} = h(\vartheta_2 - 0). $$

The variables $U_{\varphi}, V_{\varphi}$, defining the vibrations of tyre about the steady-state motion, satisfy following system

$$ -2\rho r^3 \Omega V_{\varphi}^2 + 2\rho r^3 \Omega^2 V_{\varphi} - \rho \dot{\theta}^2 U_{\varphi} $$

$$ + g_0(U_{\varphi} + V_{\varphi}) + (n_11 - n_0)U_{\varphi} - n_01U_{\varphi} + (m_20 - m_21 + n_0)V_{\varphi} + \lambda_{\varphi} = 0 $$

$$ -2\rho r^3 \Omega V_{\varphi}^2 + 2\rho r^3 \Omega^2 V_{\varphi} - \rho \dot{\theta}^2 U_{\varphi} $$

$$ - g_0(U_{\varphi} + V_{\varphi}) - (m_20 - m_21 - n_0)U_{\varphi} + n_12V_{\varphi} + (n_0 - n_02)V_{\varphi} - \lambda_{\varphi} = 0 $$

$$ 2(U_{\varphi} + V_{\varphi}) + (U_{\varphi} + V_{\varphi})^2 + (V_{\varphi} - U_{\varphi})^2 = 0 \implies U_{\varphi} = -V_{\varphi} $$

$$ (-1)^k \left[ \rho \dot{\theta} \Omega \lambda_{\varphi} - (n_0 - g_0)V_{\varphi} + (n_0 - n_1)U_{\varphi} \right]_{l(k)} + r\mu_{1k} \alpha_{\varphi} = 0 $$

$$ (-1)^k \left[ \rho \dot{\theta} \Omega \lambda_{\varphi} - (n_12 - g_0)V_{\varphi} + \lambda_{\varphi} \right]_{l(k)} + r\mu_{1k} \alpha_{\varphi} = 0, \quad [U_{\varphi}]_{k} = [V_{\varphi}]_{k} = 0, \quad k = 1, 2 $$. 

1370
Here taking into account that Lagrange multipliers $\lambda_{\text{vib}}, \mu_{\text{vib}1k}$, as well as, $\alpha_{\text{vib}k}$ are of first order of smallness is necessary. We have also used the following property $[U'_{\text{vib}}]_{l(k)} = (-1)^k [U'_{\text{vib}}]_{l(k)}$ (the functions determining the vibrations are equal to zero in the contact area). Differentiating both sides of the first equation and adding it to the second equation, using the linearised condition of the inextensibility of $l_0$, we obtain the equation and the boundary conditions

$$\rho r^3 \dddot{V}_{\text{vib}} - \rho r^3 \dddot{V}_{\text{vib}} + 2 \rho r^3 \Omega \dot{V}_{\text{vib}} + 2 \rho r^3 \Omega \dot{V}_{\text{vib}} + a_0 V^{(4)}_{\text{vib}} + a_1 V''_{\text{vib}} + a_2 V_{\text{vib}} = 0$$

$$V_{\text{vib}}(\alpha_1^0 + 2\pi + \alpha_{\text{vib}1}) = 0, V'_{\text{vib}}(\alpha_1^0 + 2\pi + \alpha_{\text{vib}1}) = 0, V_{\text{vib}}(\alpha_2^0 + \alpha_{\text{vib}2}) = 0, V'_{\text{vib}}(\alpha_2^0 + \alpha_{\text{vib}2}) = 0$$

(10)

Taking into account the dynamic boundary conditions (the fourth relation in (9)) one can determine $\alpha_{\text{vib}k}$

$$\alpha_{\text{vib}k} = \frac{(-1)^k}{r \mu_{\text{vib}1k}^2} (n_0 - n_{11}) [V''_{\text{vib}}]_{l(k)}$$

Thus, the boundaries of the contact area vibrate at the same frequency as the function $V''_{\text{vib}}$. However, in determining the frequency of vibrations of the tyre, the length of the contact area can be taken as constant, since within the model chosen its variation determines a second order of smallness correction to the frequency. Hence, the boundary conditions in problem (10) are equivalent to the following

$$V_{\text{vib}}(\alpha_1^0 + 2\pi) + V'_{\text{vib}}(\alpha_1^0 + 2\pi) \alpha_{\text{vib}1} \approx V_{\text{vib}}(\alpha_1^0 + 2\pi) = 0, V_{\text{vib}}(\alpha_2^0) + V'_{\text{vib}}(\alpha_2^0) \alpha_{\text{vib}2} \approx V_{\text{vib}}(\alpha_2^0) = 0$$

$$V_{\text{vib}}(\alpha_1^0 + 2\pi) + V'_{\text{vib}}(\alpha_1^0 + 2\pi) \alpha_{\text{vib}1} \approx V_{\text{vib}}(\alpha_1^0 + 2\pi) = 0, V_{\text{vib}}(\alpha_2^0) + V'_{\text{vib}}(\alpha_2^0) \alpha_{\text{vib}2} \approx V_{\text{vib}}(\alpha_2^0) = 0$$

For simplicity, we will write $\alpha_k$ instead of $\alpha_{\text{vib}k}^0$.

The Lagrange multipliers $\lambda_{\text{vib}}, \mu_{\text{vib}1k}$ determining, respectively, the correction to the tension and the corrections to the longitudinal reactions at the boundary points of the contact area during vibrations, are expressed from the system (9) in terms of $V_{\text{vib}}$

$$\lambda_{\text{vib}} = -\rho r^3 \dddot{V}_{\text{vib}} - 2 \rho r^3 \Omega \dot{V}_{\text{vib}} + (n_{11} - n_0) V^{(3)}_{\text{vib}} - (m_{21} - m_2 + n_0 + n_{01}) V'_{\text{vib}}$$

$$\mu_{\text{vib}1k} = (-1)^k \left[ 2 \rho r^3 \Omega \dddot{V}_{\text{vib}} + (n_0 - n_{11}) V^{(3)}_{\text{vib}} \right]_{l(k)}$$

We will represent the functions determining the vibrations in the form

$$V_{\text{vib}} = e^{i\alpha t} X(\alpha), \quad U_{\text{vib}} = -V'_{\text{vib}} = -e^{i\alpha t} X'(\alpha)$$

Substituting the expression for $V_{\text{vib}}$ into equation (10), we obtain the equation

$$a_0 X^{(4)} + 2 \rho r^3 \Omega \omega i X^{(3)} + (a_1 - \rho r^3 \omega^2) X'' + 2 \rho r^3 \Omega \omega i X' + (a_2 + \rho r^3 \omega^2) X = 0$$

with solution of form (6) (replacing $\varphi$ on $\alpha$). The characteristic equation reads

$$a_0 p^4 + 2 \rho r^3 \Omega \omega i p^3 + (a_1 - \rho r^3 \omega^2) p^2 + 2 \rho r^3 \Omega \omega i p + (a_2 + \rho r^3 \omega^2) = 0$$

(11)

The coefficients $G_i$ in the solution (6) are determined from the boundary conditions

$$G_1 e^{p_{1}(\alpha_1^0 + 2\pi)} + G_2 e^{p_{1}(\alpha_1^0 + 2\pi)} + G_3 e^{p_{1}(\alpha_1^0 + 2\pi)} + G_4 e^{p_{4}(\alpha_1^0 + 2\pi)} = 0$$

$$G_1 e^{p_{1}(\alpha_2^0)} + G_2 e^{p_{1}(\alpha_2^0)} + G_3 e^{p_{3}(\alpha_1^0)} + G_4 e^{p_{4}(\alpha_2^0)} = 0$$

$$G_1 p_1 e^{p_{1}(\alpha_1^0 + 2\pi)} + G_2 p_2 e^{p_{1}(\alpha_1^0 + 2\pi)} + G_3 p_3 e^{p_{1}(\alpha_1^0 + 2\pi)} + G_4 p_4 e^{p_{1}(\alpha_1^0 + 2\pi)} = 0$$

(12)

The homogeneous system (12) has a non-zero solution if its determinant is equal to zero:

$$f(\omega) = e^{p_{1} + p_{2} + p_{3} + p_{4} + (\alpha_1 + 2\pi)} \left[ (p_3 - p_1) (p_4 - p_2) \left( e^{p_{4} + p_{3} + (\Delta\alpha - 2\pi)} + e^{p_{4} + p_{3} + (\Delta\alpha - 2\pi)} \right) \right.$$

$$\left. - (p_3 - p_2) (p_4 - p_1) \left( e^{p_{1} + p_{2} + (\Delta\alpha - 2\pi)} + e^{p_{1} + p_{2} + (\Delta\alpha - 2\pi)} \right) \right.$$
Here $\Delta \alpha = \alpha_2 - \alpha_1$ determines the length of the contact area.

The spectrum of NF is determined from equation (13), where $p_1, p_2, p_3, p_4$ are the roots of (11).

This quartic equation can be solved by Ferrari’s method.

---

**Figure 3.** Loaded rotating tyre (Input I, $\Delta \alpha = 0.3$ rad): the NF as a function of angular velocity, (a) Lagrangian specification, (b), (c) Eulerian specification.
Figure 4. The rotation of the second MS ($\nu_2 = 94.93$ Hz) of a loaded rotating tyre (Input $I$, $\Delta \alpha = 0.3$ rad, $\Omega = 175$ rad s$^{-1}$).

\[ p_{1,2}(\omega) = \frac{1}{2} \left( \sqrt{2\gamma_0} \pm \sqrt{-2(\gamma_0 + A_1) - \frac{2A_2}{\sqrt{2\gamma_0}} \frac{\rho r^3 \Omega \omega}{a_0} i} \right) \]
\[ p_{3,4}(\omega) = \frac{1}{2} \left( -\sqrt{2\gamma_0} \pm \sqrt{-2(\gamma_0 + A_1) + \frac{2A_2}{\sqrt{2\gamma_0}} \frac{\rho r^3 \Omega \omega}{a_0} i} \right) \]

\[ \gamma_0 = -\frac{A_1}{3} + C_1 + C_2, \quad C_{1,2} = \sqrt{-\frac{B_2}{2} \pm \sqrt{\frac{B_2^2}{4} + \frac{B_1^3}{27}}}, \quad B_1 = -A_3 - \frac{A_2^2}{12}, \quad B_2 = A_1 A_3 - \frac{A_2}{8} \frac{A_1^3}{108} \]
\[ A_1 = \frac{2a_0(a_1 - \rho r^3 \omega^2) + 3g_0 \rho r^3 \omega^2}{2a_0^2}, \quad A_2 = \frac{\rho r^3 \Omega \omega(2a_0^2 - a_0(a_1 - \rho r^3 \omega^2) - g_0 \rho r^3 \omega^2)}{a_0^4} \]
\[ A_3 = \frac{16a_0^3(a_2 + \rho r^3 \omega^2) + 4a_0 g_0 \rho r^3 \omega^2(4a_0 - a_1 + \rho r^3 \omega^2) - 3g_0^2(\rho r^3 \omega^2)^2}{16a_0^4} \]

Since all nonzero complex numbers have three distinct complex cube roots, then the third relation in (14) gives three values for $\gamma_0$. One can take any of these three values, but one cannot combine arbitrary value of the first cube root $C_1$ with another arbitrary value of the second cube root $C_2$. If some value for the first cube root was chosen, then one must chose the value for the second cube root such that $C_1 C_2 = -B_1 / 3$. 

1373
The quantities of obtained NF \(v_n = \omega_n/(2\pi)\) of a loaded non-rotating tyre were compared [10] with results of experiment [11]. The plot of the first twenty NF as a function of angular velocity for the Input I, \(\Delta\alpha = 0.3\) rad is shown in Figure 3 (b). The figure shows that an increasing of the angular velocity implies that NF decrease, in agreement with results of [12], where FE-based approach was used. The previously observed split of NF of an unloaded rotating tyre caused by rotation disappears under rolling conditions due to the disturbed symmetry, that also agrees with the results of [12]. In addition, an interesting phenomenon of frequency loci veering (Figure 3 (c)) is visible here: the NF-lines approach each other and suddenly veer away instead of crossing [5]. One can note that the MS interact in frequency loci veering region and finally interchange.

**Remark 2.** Using the Lagrangian specification the term of the third degree \(p^3\) disappears (the characteristic equation (7)). Then, taking into account the Viète’s formulæ \(p_4^{(3)} = -(p_1 + p_2 + p_3)\), the determinant (13) can be represented as

\[
\frac{f(\omega)}{2} = \left[ (p_1 + p_2)^2 - (p_2 + p_3)^2 \right] \text{ch} \left( (p_1 + p_3)(2\pi - \Delta\phi) \right) + \left[ (p_1 + p_3)^2 - (p_1 + p_2)^2 \right] \text{ch} \left( (p_2 + p_3)(2\pi - \Delta\phi) \right) + \left[ (p_2 + p_3)^2 - (p_1 + p_2)^2 \right] \text{ch} \left( (p_1 + p_2)(2\pi - \Delta\phi) \right) = 0
\]

where \(\Delta\phi = \phi_2 - \phi_1\) determines the length of the contact area. The spectrum of NF reads

\[
p_{1,2}(\omega) = \frac{1}{2} \left( \sqrt{2\gamma_0 \pm \left[ -2(\gamma_0 + A_1) - \frac{2A_2}{\sqrt{2\gamma_0}} \right]} \right), \quad A_1 = \frac{a_1 - 3g_0 - \rho r^3 \omega^2}{a_0 - g_0} \]

\[
p_{3,4}(\omega) = \frac{1}{2} \left( -\sqrt{2\gamma_0 \mp \left[ -2(\gamma_0 + A_1) - \frac{2A_2}{\sqrt{2\gamma_0}} \right]} \right), \quad A_2 = \frac{4\rho^2 \Omega \omega}{a_0 - g_0} i, \quad A_3 = \frac{a_2 + \rho r^3 \omega^2}{a_0 - g_0}
\]

The plot of the first thirteen NF as a function of angular velocity is shown in Figure 3 (a). The MS reads

\[X(\alpha) = G_1 e^{p_1 \alpha} + G_2 e^{p_2 \alpha} + G_3 e^{p_3 \alpha} + G_4 e^{p_4 \alpha}\]

where the constants \(G_i\) are determined from boundary conditions (12)

\[G_1 = e^{-p_1(\alpha_1 + 2\pi)} \left( (p_4 - p_3) e^{p_4(\Delta\alpha - 2\pi)} - (p_4 - p_2) e^{p_2(\Delta\alpha - 2\pi)} + (p_3 - p_2) e^{p_2(\Delta\alpha - 2\pi)} \right) G_5^*\]

\[G_2 = e^{-p_2(\alpha_1 + 2\pi)} \left( (p_4 - p_3) e^{p_4(\Delta\alpha - 2\pi)} + (p_4 - p_1) e^{p_1(\Delta\alpha - 2\pi)} - (p_3 - p_1) e^{p_1(\Delta\alpha - 2\pi)} \right) G_5^*\]

\[G_3 = e^{-p_3(\alpha_1 + 2\pi)} \left( (p_4 - p_2) e^{p_4(\Delta\alpha - 2\pi)} - (p_4 - p_1) e^{p_1(\Delta\alpha - 2\pi)} + (p_2 - p_1) e^{p_1(\Delta\alpha - 2\pi)} \right) G_5^*\]

\[G_4 = e^{-p_4(\alpha_1 + 2\pi)} \left( (p_3 - p_2) e^{p_2(\Delta\alpha - 2\pi)} + (p_3 - p_1) e^{p_1(\Delta\alpha - 2\pi)} - (p_2 - p_1) e^{p_1(\Delta\alpha - 2\pi)} \right) G_5^*, \forall G_5^*
\]

The function \(V_{ vib}(\alpha, t)\) is represented by the linear combination

\[V_{ vib}(\alpha, t) = \left( \cos(\omega t) \text{Re}(X(\alpha)) - \sin(\omega t) \text{Im}(X(\alpha)) \right) + i \left( \cos(\omega t) \text{Im}(X(\alpha)) + \sin(\omega t) \text{Re}(X(\alpha)) \right)\]

of real MS \(\text{Re}(X(\alpha))\) and of imaginary MS \(\text{Im}(X(\alpha))\) corresponding to the same NF. Eventually the real MS is transformed into imaginary MS and vice versa

\[V_{ vib}(\alpha, 0) = \text{Re}(X(\alpha)) + i \text{Im}(X(\alpha)), \quad V_{ vib}(\alpha, \frac{\pi}{2\omega}) = -i \text{Im}(X(\alpha)) + i \text{Re}(X(\alpha))\]

The rotation of the second MS \((v_2 = 94.93\) Hz) is represented in Figure 4 for the Input I, \(\Delta\alpha = 0.3\) rad, \(\Omega = 175\) rad-s\(^{-1}\). The rolling occurs in the clockwise direction. The evolution of the third real and imaginary MS of a loaded rotating tyre (Input I, \(\Delta\alpha = 0.3\) rad) is represented in Figure 5. For \(\Omega = 3, 100, 130\) rad-s\(^{-1}\) the third MS of a loaded rotating tyre has, respectively, three, four, five nodes and is similar to the third, fourth, fifth MS of a loaded non-rotating tyre. Thus, the third MS changes from a three-node to a five-node shape, while the NF decreases from 116.91 Hz to 102.81 Hz.
Figure 5. The evolution of the third real (———) and imaginary (——) MS of a loaded rotating tyre (Input I, $\Delta \alpha = 0.3$ rad), (a) $\Omega = 3 \text{ rad} \cdot \text{s}^{-1}$, $v_3 = 116.91 \text{ Hz}$, (b) $\Omega = 100 \text{ rad} \cdot \text{s}^{-1}$, $v_3 = 107.71 \text{ Hz}$, (c) $\Omega = 130 \text{ rad} \cdot \text{s}^{-1}$, $v_3 = 102.81 \text{ Hz}$.

5 CONCLUSIONS

We have thus completed the investigation of vibrations of an unloaded rotating and loaded rotating tyre rolling at constant speed without slipping in the contact area. The NF and MS of an unloaded and loaded tyre are determined. Each NF of an unloaded non-rotating tyre corresponds to two NF of an unloaded rotating tyre. This is a well known effect: the two counter-rotating waves, superimposed onto a standing vibration in resting structures, are distinct in speed for a rotating system.

In the case of loaded rotating tyre, in determining the frequency of the vibrations, the length of the contact area was taken as constant, since its variation determines a second order of smallness correction to the frequency, in the model chosen. The Lagrange multipliers, determining the correction to the tension of the median line of the tread and the corrections to the longitudinal reactions at the boundary points of the contact area during vibrations are found. The increasing of the angular velocity implies that NF decrease. The previously observed split of NF of an unloaded rotating tyre caused by rotation disappears under rolling conditions due to the disturbed symmetry. In addition, an interesting phenomenon of frequency loci veering is visible here: NF-lines approach each other and suddenly veer away instead of crossing. The MS interact in frequency loci veering region and finally interchange.

Acknowledgements

This investigation was fulfilled, in particular, in the Nizhny Novgorod State Technical University n.a. R.E. Alekseev (NNSTU) within the State support, provided by Minobrnauka of the Russian Federation (project identifier RFMEFI57714X0080).

A MODEL PARAMETERS

Figure 6. The geometric parameters of the undeformed tyre.
Table 1. The input data of the model (Input I).

<table>
<thead>
<tr>
<th>$m$ (kg)</th>
<th>$p$ (Pa)</th>
<th>$r_1$ (m)</th>
<th>$a$ (m)</th>
<th>$l_1$ (m)</th>
<th>$l$ (m)</th>
<th>$r$ (m)</th>
<th>$b$ (m)</th>
<th>$c$ (m)</th>
<th>$\psi_1$ (rad)</th>
<th>$\psi_2$ (rad)</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.37</td>
<td>250000</td>
<td>0.2032</td>
<td>0.09737</td>
<td>0.1025</td>
<td>0.0875</td>
<td>0.3169</td>
<td>0.1773</td>
<td>0.2455</td>
<td>1.1565</td>
<td>1.812</td>
</tr>
</tbody>
</table>

The geometric parameters of the tyre are represented in Figure 6. The point K is the centre of a circle of radius $b$, an arc of which is the sidewall. The values used for model parameters are represented in Table 1. The parameters $b$, $c$, $\psi_1$, $\psi_2$ depend on the parameters $r_1$, $a$, $l_1$, $l$, $r$.

REFERENCES


Virtual Articulated Haulers -
prediction of vehicle performance and design loads

Torbjörn Ekevid*, Conny Carlqvist# and Therese Sjöden*

* Volvo Construction Equipment AB
Vehicle System Simulations
Carl Lihnells väg 2, 360 42 Braås, Sweden
[torbjorn.ekevid,therese.sjoden]@volvo.com

# Volvo Construction Equipment AB
Vehicle System Simulations
Bolindervägen 5, 635 10 Eskilstuna, Sweden
conny.g.carlqvist@volvo.com

ABSTRACT
Articulated haulers are suited for any kind of transports in difficult terrain. The concept was
developed from an agricultural tractor in the 1960’s by Volvo. The main features are the hydraulic
steering, the individual front and rear frames connected with a rotational hitch joint, all-wheel
drive and large tires. These features give the extraordinary terrain capability. Complete vehicle
simulations support development of stability, handling and ride comfort properties during early
development of the hauler. Complete vehicle simulations can also be utilized to extract design
loads. This paper presents the current status in this field.
The complete vehicle MBS simulation model includes flexible bodies, hydraulic system, tires
and powertrain. A refined powertrain model in Simulink is used for the analysis presented. The
environment is defined by the road and the driver. A comfort track road is used for comfort
evaluations and a digitalized endurance test road for load predictions. The driver model aims to
follow the given target path and velocity.

Simulation and measurement signals are compared by means of relevant spectra; load ranges,
level crossings and Power Spectral Density. A scalar measure is defined by the equivalent value
to carry out comparisons between simulations and test. Fatigue load simulations need an accurate
dynamic description of the vehicle, capturing low frequency excitation. The results shows that the
load simulation captures the frequency peaks below 15 Hz, although there are some differences
in magnitude and peak frequency. When simulated and measured equivalent values are compared
most entities are in the range between 1/2 and 2 regarding estimated life. Further improvements are
possible but the simulated load quality should also be related to other uncertainties in the fatigue
evaluation process.

Keywords: Articulated Hauler, Co-simulation, Structure loads, Multi Body Dynamics.

1 INTRODUCTION
In product development of construction equipment, complete vehicle simulation models are used
for a rapid and robust product development process. Many properties e.g. stability, handling
and ride comfort is secured with high accuracy early in the product development process. Sev-
eral sizes and configurations of a particular vehicle type e.g. a hauler, must be available on the
market to cover most of the expectations from costumers. As a consequence, a relatively small
number of vehicles of a certain configuration will be produced during the lifetime of each gener-
atation. This is one of the main reasons why construction equipment manufacturers did not adopt
simulation-driven development before, but to large extent relied on old-fashion test-driven product
development.

Although a lot of the verification can be done by means of simulations, testing activities are still
very important. To have confidence in simulations, models and/or component data have to be
updated and verified continuously. Such data includes mass and inertia properties for the flexible or
rigid components, stiffness and damping characteristics of flexible elements, tire properties, engine
characteristics etc. By a simulation-driven, development the testing activities do not decrease but
change focus from complete vehicle testing to component testing and virtual model correlation testing.

At Volvo Construction Equipment, MBS models of haulers have been used for more than 15 years. However, during the last couple of years the MBS models are also utilized to extract design loads. This paper presents the current status in this field. The paper is organized as follows: First a brief introduction of the articulated hauler concept is given. Section 2 describes the full vehicle model and highlights some modeling features e.g. hydraulic and powertrain models. In section 3, some aspects regarding the environment i.e. the road and the driver is discussed. Section 4 describes the evaluation procedures of the simulation and test results presented in section 5. The paper ends with some concluding remarks and proposals for further research and development in section 6.

1.1 Articulated haulers

Articulated haulers are suited for any kind of transports in difficult terrain. From initially being designed as a earth-moving vehicle, the vehicle can be used as a load carrier for pipes, timber, containers etc. Figure 1 shows an articulated hauler in a typical application.

![Figure 1. Articulated hauler in rough terrain application.](image)

The hauler concept was developed in the 1960’s by Volvo. It originates from a conventional agricultural tractor coupled to a driven load unit with a hydraulic dump body. Although the vehicles have undergone substantial evolution since that time, the main features giving the vehicles extraordinary terrain capability remains. These are the hydraulic articulated steering, the individual front and rear frames connected with a rotational hitch joint and all-wheel drive together with large tires. Today, the vehicles are also equipped with automatic differential locks, an automatic powershift transmission and wet brakes with external cooling.

2 THE HAULER MODEL

Since approximately 5 years, the hauler simulation environment at Volvo CE makes use of features in ADAMS/Car[1]. Before adopting the models to ADAMS/Car, the assembly and simulation of hauler models were done in an in-house application added to the ADAMS/view interface.

2.1 Vehicle assembly

The hauler model is assembled from different subsystems. Each subsystem is based on a template which specifies the topology of the subsystem i.e. the different components in the subsystem and how they interact. The template also specifies how the subsystem should communicate or connect to other subsystems. Usage of minor roles makes it possible to use the same template for several subsystems eg. tires, axles. Figure 2 shows the hauler model of an A40, where some of the exterior components have been removed from the model to be able to study the frame structure in details. The mass and the inertia of these components are included in the mass properties of the frames.
2.1.1 Flexible bodies

For key components, vital for the dynamic properties of the vehicle, flexible bodies have been used. ADAMS makes use of Craig-Bampton reduction and an additional orthogonalization step in order to eliminate the interior deformation due to unit displacements at the interface nodes from the formulation. As a result, the interior deformation is expressed only by a set of modal coordinates superimposed by the corresponding C-B mode shapes. In figure 3, two essential C-B modes of the front frame are shown.

![Figure 3. Flexible modes of the front frame.](image)

(a) 1st torsion mode  
(b) 1st bending mode

2.2 Hydraulic system

The hydraulic steering and suspension system is modeled in LMS Amesim[2]. The hydraulic suspension system is offered as option for the large hauler platform(A35/40). The system consists of a hydraulic cylinder working against a pre-charged piston accumulator. The accumulator pre-charge pressure in conjunction with the cylinder and piston area determines the spring characteristics whereas orifices controls the damping properties. The left and the right cylinders are cross coupled for a hydraulic anti-roll function i.e. increased rolling stiffness. Further, a control system adjusts the hydraulic valves to keep the ride height and suppress rolling motions.

Figure 4 shows the hydraulic system model. The interface block in the center is used to communicate with the ADAMS solver in co-simulation mode through a general state equation.
2.3 Tires

Tire is a crucial component for a real hauler and so is the tire model for a virtual vehicle model. A number of tire models are available, and the model should be selected from the purpose of the analysis. For handling maneuvers and stability testing, tire models based on magic formula[3] is sufficient. However, for virtual load prediction, a physical tire model e.g. FTire[4] or CDTire[5] is necessary to resolve short wave length road excitation with multi-point contact. Figure 5 shows the detailed deformation of the FTire model when passing a cleat and simultaneously being subjected to a lateral force.

Compared to tires for passenger cars or trucks, where the parameters can be determined from physical testing in rigs, the parameters for large construction equipment tires must be obtained from finite element models since testing in rigs are not possible although desired. Consequently, close
co-operation with tire manufactures are important for the construction equipment manufactures.

2.4 Powertrain

The main object of a hauler driveline is the engine and gearbox assembly. At present stage, there are two options when it comes to modeling this system. The simplest option is to make use of the engine/transmission model available in ADAMS/Car. Although the real powertrain makes use of a torque converter with lock-up and an automatic planetary gearbox with powershift, the manual gearbox option with short gear shifting time can be used since the lockup function is engaged almost 100% of the time during normal driving conditions. The engine characteristic is just modeled by a throttle-rpm-torque relation.

A refined engine and gearbox model are used when required e.g. for studies of loads on driveline components downstream the gearbox. The powertrain model is composed from component models implemented into basic Simulink, see Figure 6. In this model, the automatic gear shifting sequence including the lockup functionality and the planetary gear box is more accurately modeled. The execution of the co-simulation is governed by Simulink since the ADAMS model is introduced as a s-function by a plant export.

![Simulink powertrain model](image)

The driveline downstream the gear box interface is modeled by rigid components and torsional spring/damper elements to account for at least the first couple of torsional modes of the propulsion and drive shafts. The axle center gear and the hub reduction planetary gear are simplified and just modeled by reduction gear ratios. Inside the axle and the transfer gear box, the differentials are modeled and distribute the applied torque equally between the wheels of an axle or between the front axle and the rear bogie axle. The differential locks are modeled by a user function which could be activated through a control system if necessary.

3 ENVIRONMENT

In this context, the road on which the vehicle is running and the driver that controls the vehicle represents the environment.

3.1 Roads

A cast concrete track with a well-defined track profile is used for comfort evaluations. The track has its corresponding virtual representation and is used as a reference, when evaluating virtual hauler models.

For virtual load prediction, the description of the road surface is almost as important as the tire models. Simple road descriptions may be sufficient for handling and for stability studies, but for
load prediction, an accurate representation of the road surface is required. Figure 7(a) shows the virtual endurance test track which was digitalized by means of a laser scanning equipment in 2011. The total length of the track is approximately 1120 m and the height difference between the highest and the lowest point is approximately 3.5 m.

The original road data has been processed and converted to the OpenCRG[6] format where the spatial resolution is approximately two centimeters, see figure 7(b). The virtual road is modeled as rigid which may be sufficient since the track surface consists of densified gravel.

![Figure 7. Virtual Braås II endurance test track.](image)

### 3.2 Driver

The simulations make use of the driver model available in ADAMS/Car. It uses a virtual gyro mounted on the machine as input and generates output signals that controls throttle, break, steering etc. The gyro defines the path error i.e. the distance between the path and the gyro, used to control the steering wheel. For cars with Ackermann steering, the gyro is by default located between the rear wheels of a vehicle. This position is not suitable for an articulated vehicle. Instead the gyro is positioned between the front wheels for better stability and control of the vehicle.

#### 3.2.1 Target path and velocity profile

The target path is extracted from the road information, the size of vehicle and the steering performance. The target velocity profile is predicted by a simplified vehicle model running along the track and taking the slope, banking and the curvature of the road into account. Figure 8 shows the predicted target velocity profile (blue) on the endurance test track, compared to a measurement (red) for one lap.

![Figure 8. Predicted target velocity compared to measured velocity.](image)
4 EVALUATION OF RESULTS
Comparing signals from simulations and measurements is a delicate task. Since the simulation model is controlled by a closed-loop driver, the vehicle is not able to keep exactly the same velocity as in the measurements. Instead of measures based on sample by sample comparison, the evaluation here is done by means of relevant spectra.

4.1 Fatigue spectra
High-cycle fatigue evaluation is normally based on load ranges and the number of corresponding occurrences. To obtain this information, rainflow counting [7] is applied to the time signal. For some materials the mean-stress is important for fatigue life and level-crossing calculation is used for this purpose. The fatigue spectral information is normalized to one hour to take care of the time signal length. Power Spectral Density (PSD) visualizes the frequency content of the signals. For acceleration signals, the fatigue spectra is not very relevant, since the result depends highly on the sampling frequency.

Figure 9 shows time signal and the spectral information for one transducer measuring bending moment of the front frame. There is an offset in the time signals since the gravity forces are included in the simulations (blue) while the measurement results (red) just consider changes from static equilibrium. The reader should also observe the phase offset after some time, which is a result of small differences in vehicle speed.

4.2 Scalar fatigue measure
To make the information even more comprehensive, a scalar measures related to fatigue is defined. Let \( \{ (n_i, R_i) \} \) be the number of cycles \( n_i \) larger than the corresponding range \( R_i \) computed by the rainflow counting using \( N \) different classes. The Duty per hour becomes

\[
D = k_t \sum_{i=1}^{N} n_i R_i^m,
\]

where \( k_t \) is a factor that normalizes the Duty to one hour and \( m \) is the Whöler exponent. The equivalent range i.e. the constant range loading at 1000 cycles per hour that gives the same Duty per hour as the original signal is defined by

![Figure 9. Time signals and different spectra.](image-url)
\[ R_{eqv} = \left( \frac{D}{1000} \right)^{1/m} \]. \tag{2} 

The comparison between simulations and test results below uses the relative equivalent values

\[ \mu_i = \frac{\hat{R}_{eqv,i}}{\bar{R}_{eqv}} \] \tag{3}

i.e. the quotient between equivalent value \( \hat{R}_{eqv,i} \) from the tests and the mean (most cases just one) of the simulated equivalent values \( \bar{R}_{eqv} \) for a certain test track.

5 RESULTS

This section presents results from a comparison between simulation and several measurements.

5.1 Instrumentation during test

A combination of accelerometers and calibrated strain gauges glued at relevant positions are used for a load measurements on a haulers. In figure 10, the instrumentation during one recent test is given. Typical entities measured during the test are: frame bending moments (XX_MBY_NN), cross stay forces (XX_FY), axle stay moments (XX_MBY) and forces (XX_FX, XX_FX), bogie beam vertical forces (BB_5F_FZ). Also accelerations (red text) on the left hand side of the axles (VXX_Y, VXX_Z) and on the cabin floor (Seat_Z) are measured.

![Figure 10. Instrumentation of the hauler.](image)

5.2 Comfort test track

On the comfort test track the test driver follows a certain path at constant velocity. The gearbox is prevented to change gear during the test and the throttle position is controlled by an external device. The results presented in this section comes from a test with a target velocity of 8 km/h running on the path with 8 cm obstacles.

To show some details, the vertical acceleration measured on top side of the axle casting (VFA_Z) is shown in figure 11(a), while figure 11(b) shows the acceleration at the cabin floor position (Seat_Z). The agreement between test (red) and the simulation is fairly good, but at the front axle position there are some differences around 6-7 Hz where there is a peak in the simulations that is not present in the test result. For the cabin floor position, these frequencies are damped out by the
front axle suspension. The bending moment at position FR_MBY_11 for the same setup is given in figure 9.

To summarize the comparison for all transducers available, figure 12 is used. The figure shows the simulation result normalized to unity (blue diamonds) and the relative equivalent value \( \mu_i \) (red asterisk) for the test. The green lines represent a relative difference in fatigue loading that corresponds to an expected difference in life of 1/2 and 2 for a material with the same Whöler exponent as the equivalent values were computed.

![Figure 11. Tractor unit vertical acceleration.](image)

![Figure 12. Simulation and test result at the Comfort test track.](image)

5.3 Endurance test track

During a measurement on the endurance test track, the vehicle runs four laps counter-clock wise (CCW) and four laps clock wise (CW). There are no other instructions to the driver but to follow the track and run the vehicle as fast as possible. One lap takes approximately 185 s to complete if initially starting from rest.

Figure 13 and figure 14 shows the comparison between one simulation and several test laps. For each direction (CW or CCW), the first three laps are considered and asterisks with red color corresponds to a test recently carried out, while the cyan corresponds to the test which was carried out a few years ago. The figures also give some ideas of a prediction intervals for the different locations although just two tests are considered. If studying each location more closely, there are some differences in the results when it comes to frequency content. For frequencies below 15 Hz, most of the peaks in simulation PSD coincides well with the peaks in tests but there are also
differences both when it comes to peak magnitude and the corresponding frequency.

For the load carrying structure on a hauler, most of the fatigue damage of a component is governed by low frequency loads and its corresponding stresses. The low frequency loads is typically induced by road unevenness and vehicle maneuvers while the magnitude of the response is determined by the dynamics of the vehicle. Consequently, to simulate hauler fatigue loads it is especially important to have an accurate dynamic description of the vehicle for low frequency excitation.

6 CONCLUDING REMARKS AND PROPOSAL OF FURTHER RESEARCH

A number of positions and entities have been considered when evaluating simulation results against tests. For most positions and physical entities, the simulations are in the range between 1/2 and 2 compared to test result regarding estimated life. There is potential to further improve the simulation models, but the quality of the simulation results should be related to the uncertainties in estimating customer usage, the FE-modeling of welded structures and and the spread in material fatigue capacity.

There is a continuous development of the simulation models for performance and load predictions going on. Components that today are modeled as rigid e.g. the longitudinal Axle-stays, should be replaced by flexible components to increase the fidelity. Also limited functionality of the system
responsible for drive line control e.g. differential locks and the brakes should be included since it is vital for some driving cases. The models are at present time running on rigid surface roads, but there is a need to include the interaction between the tire and the flexible ground having a realistic rheological model of the ground material. This would extend the usage of the models to also cover refined studies of traction, gradeability and fuel consumption.

REFERENCES
Development of an Energy-Saving Manipulator Using Storage Elements and Reaction Wheels

Makoto Iwamura*, Shunichi Imafuku*, Takahiro Kawamoto*, Werner Schiehlen#

*Department of Mechanical Engineering
Fukuoka University
Jonan, 814-0180 Fukuoka, Japan
iwamura@fukuoka-u.ac.jp

#Institute of Engineering and Computational Mechanics, University of Stuttgart
Pfaffenwaldring 9, 70569 Stuttgart, Germany
schiehlen@itm.uni-stuttgart.de

ABSTRACT
In this paper, we propose a concept for the design and control of an energy saving manipulator utilizing springs and reaction wheels. Firstly, we examine the simultaneous optimization problem of spring parameters and trajectories with respect to the energy consumption based on optimal control theory. We analyze the relationship between the consumed energy and the operating time, derive a condition for the operating time to be optimal, and propose a corresponding design method for springs. After that, we consider the practical design problem to realize the proposed energy saving manipulator concept. In order to verify the proposed method, a prototype 2DOF manipulator is developed by using linear springs and reaction wheels. The experimental results show the effectiveness of the proposed energy saving manipulator concept.

Keywords: Robot Manipulators, Energy Saving, Springs, Reaction Wheels, Optimal Control.

1 INTRODUCTION
In the manufacturing industry, machines and robots controlled by actuators are used to increase productivity and achieve high quality products. However, these actuators are consuming a great amount of energy accelerating and braking continuously. Hence, saving the energy of such mechanical systems is a very important issue.

Methods for reducing the consumed energy of controlled multibody systems by utilizing passive storage elements such as springs have been recently examined, see Ref.[1] to [4]. In this paper, we consider the case of a planar NDOF serial manipulator with springs as shown in Figure 1. Firstly, we analyze theoretically how to choose the spring stiffnesses, spring mounting positions, and moving robot trajectories to maximize energy savings. Based on the results of the analysis, we propose a simultaneous optimization method of spring parameters and trajectories. We show the effectiveness of the proposed method by numerical simulations.

In theory, we can achieve zero energy consumption by the proposed method, but in practice the method cannot be directly applied to the conventional manipulator since it utilizes the free frictionless vibrations of the system. Therefore, we next consider the practical design and control problems to realize the proposed energy saving manipulator concept. We propose a design method that utilizes the linear springs and controlled reaction wheels and develop a prototype 2DOF energy saving manipulator based on the method. Finally, the effectiveness of the proposed method is verified through laboratory experiments.

Figure 1. Planar NDOF serial manipulator with storage springs.
2 ENERGY SAVING CONTROL METHOD USING SPRINGS

This paper discusses the reduction of the energy consumption of SCARA robots by adding springs to the joints as shown in Fig. 1. In this chapter, we consider the simultaneous optimization problem of spring parameters and robot trajectories that minimizes the energy consumption based on the optimal control theory.

2.1 Problem formulation

The equations of motion of $N$-link serial horizontal manipulators equipped with springs can be expressed as follows

$$M(\theta) \ddot{\theta} + h(\theta, \dot{\theta}) = -K(\theta - \theta_n) + u,$$

where $\theta = [\theta_1, \theta_2, \cdots, \theta_N] \in \mathbb{R}^N$ is the joint variable vector, $M \in \mathbb{R}^{N \times N}$ is the inertia matrix, $h \in \mathbb{R}^N$ is the vector of centrifugal and Coriolis forces, $u = [u_1, u_2, \cdots, u_N] \in \mathbb{R}^N$ is the vector of direct driving torques, $K = \text{diag}[k_1, k_2, \cdots, k_N] \in \mathbb{R}^{N \times N}$ is the rotational stiffness matrix, $\theta_n \in \mathbb{R}^N$ is the vector of spring mounting positions. Here, we consider motions between two points and rest at both ends, i.e.,

$$\theta(0) = \theta_0, \quad \theta(t_f) = \theta_f,$$

$$(2)$$

$$\dot{\theta}(0) = 0, \quad \dot{\theta}(t_f) = 0,$$

$$(3)$$

where $\theta_0$ and $\theta_f$ are the initial and final configurations and $t = t_f$ is the final time. The energy consumed by this motion can be estimated by the following cost function

$$J = \int_0^{t_f} f_0(x, u) dt, \quad f_0(x, u) = \frac{1}{2} u^T W u,$$

$$(4)$$

where $W \in \mathbb{R}^{N \times N}$ is a positive definite symmetric matrix.

The problem considered in this section can be formulated as follows: Find $u(t), \theta(t), \theta_n$ and $k = [k_1, k_2, \cdots, k_N]^T$ by minimizing the energy consumption (4) subject to the initial and final conditions (2) and (3) for the system (1).

2.2 Theoretical analysis of minimum energy control

It is difficult to obtain analytical solutions of the problem formulated in 2.1 since the Eq. (1) has a strong nonlinearity. Hence, in this section, we analyze the problem approximately by using the linearization and modal analysis techniques.

Firstly, we shift the reference point to the middle point of the initial and final configurations in order to make the later calculations easy. We define $\theta_m = \frac{1}{2} (\theta_f + \theta_0)$ and $\theta_e = \frac{1}{2} (\theta_f - \theta_0)$, and shift the coordinates as $\tilde{\theta}(t) = \theta(t) - \theta_m, \tilde{\theta}_n = \theta_n - \theta_m$. This transforms the initial and final conditions to the symmetric form

$$\tilde{\dot{\theta}}(0) = -\theta_e, \quad \tilde{\dot{\theta}}(t_f) = \theta_e,$$

$$(5)$$

$$\tilde{\dot{\theta}}(0) = 0, \quad \tilde{\dot{\theta}}(t_f) = 0.$$

$$(6)$$

And the equations of motion (1) is transformed to the following form

$$\tilde{M}(\tilde{\theta}) \ddot{\tilde{\theta}} + \tilde{h}(\tilde{\theta}, \dot{\tilde{\theta}}) + K \tilde{\theta} = u + K \tilde{\theta}_n,$$

$$(7)$$

If it is assumed that fairly strong springs are used, the spring forces become predominant over the centrifugal and Coriolis forces. Therefore, we neglect these forces. Additionally, we assume that the inertia matrix can be approximated at the middle point $\tilde{\theta} = \theta = \theta_m$ as $\tilde{M}(\theta_m) = \tilde{M}(0) \equiv \tilde{M} = \text{const}$. Under these assumptions, the following linearized equations of motion are obtained

$$\tilde{M} \ddot{\tilde{\theta}} + K \tilde{\theta} = u + K \tilde{\theta}_n.$$

$$(8)$$
Let us consider the free vibration system corresponding to Eq. (8) and calculate the modal matrix \( \Phi \in \mathbb{R}^{N \times N} \) that satisfies
\[
\Phi^T \dot{M} \Phi = I, \quad \Phi^T K \Phi = \Omega^2, \quad \Omega = \text{diag}[\omega_1, \omega_2, \cdots, \omega_N],
\]
where \( I \) is the identity matrix, \( \omega_i \) is the \( i \)-th natural frequency arranged as \( \omega_1 < \omega_2 < \cdots < \omega_N \) where some of the lowest may vanish. We make the coordinate transformation \( q = \Phi^{-1} \tilde{\theta} \) \( (q_n = \Phi^{-1} \theta_n) \) and define the state vectors \( x = [x_1^T, x_2^T]^T = [q^T, \dot{q}^T]^T \). Then, the following state equations are found
\[
\begin{align*}
\dot{x}_1 &= x_2 (\equiv f_1(x,u)), \\
\dot{x}_2 &= -\Omega^2 x_1 + \Phi^T u + \Omega^2 q_n (\equiv f_2(x,u)).
\end{align*}
\]

The initial and final conditions are expressed as
\[
\begin{align*}
x_1(0) &= -q_e, & x_1(t_f) &= q_e, \quad (12) \\
x_2(0) &= 0, & x_2(t_f) &= 0, \quad (13)
\end{align*}
\]
where \( q_e = \Phi^{-1} \theta_e \).

Next, let us introduce an adjoint vector \( \psi = [\psi_1^T, \psi_2^T]^T \) and define the Hamiltonian as follows
\[
H = f_0 + \psi_1^T f_1 + \psi_2^T f_2 = \frac{1}{2} u^T W u + \psi_1^T x_2 + \psi_2^T (-\Omega^2 x_1 + \Phi^T u + \Omega^2 q_n). \quad (14)
\]

Then, the optimal control is derived from the condition \( \partial H / \partial u = 0 \) as
\[
u = -W^{-1} \Phi \psi_2. \quad (15)
\]

Substituting Eq. (15) into Eq. (14), the Hamiltonian along the optimal trajectory is given by
\[
H = \psi_1^T x_2 - \psi_2^T \Omega^2 x_1 - \frac{1}{2} \psi_2^T \Phi^T W^{-1} \Phi \psi_2 + \psi_2^T \Omega^2 q_n. \quad (16)
\]

From Eq. (16), the canonical equations of Hamilton can be derived as follows
\[
\begin{align*}
\dot{x} &= \partial H / \partial \psi = Ax + B \psi + c_n, \quad (17) \\
\dot{\psi} &= -\partial H / \partial x = -A^T \psi, \quad (18)
\end{align*}
\]
where
\[
A = \begin{bmatrix} 0 & I \\ -\Omega^2 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 0 \\ 0 & -\Phi^T W^{-1} \Phi \end{bmatrix}, \quad c_n = \begin{bmatrix} 0 \\ \Omega^2 q_n \end{bmatrix}. \quad (19)
\]

By solving the differential equations (17) and (18) under the boundary conditions (12) and (13), we obtain the optimal solution that minimizes the energy consumption. However, it is difficult to obtain the closed-form analytical solution of these equations since they are coupled due to the term \( \Phi^T W^{-1} \Phi \) in the matrix \( B \). In contrast, choosing the weighting matrix as \( W = \hat{M}^{-1} \) results in the cost function of mechanical power and allows to decouple these equations by the property \( \Phi^T \hat{M}^{-1} \Phi = \Phi^T \hat{M} \Phi = I \). Therefore, in the following, we analyze this case.

Let us denote the initial value of adjoint vector as \( \psi(0) \), then the solution of Eq. (18) reads
\[
\psi(t) = e^{-A_T} \psi(0). \quad (20)
\]

By using Eq. (20), the solution of Eq. (17) is derived as
\[
\begin{align*}
x(t) &= e^{A t} x(0) + e^{A t} \int_0^t e^{-A \tau} B e^{-A_T} \tau \psi(0) + e^{A t} \int_0^t e^{-A \tau} \tau c_n. \quad (21)
\end{align*}
\]
If the final condition \( x(t_f) \) is specified, \( \psi(0) \) can be computed from Eq. (21) as follows
\[
\psi(0) = \left[ \int_0^{t_f} e^{-A^t} Be^{-A^t} dt \right]^{-1} \times \left[ e^{-A^t} x(t_f) - x(0) - \int_0^{t_f} e^{-A^t} dt c_n \right].
\] (22)

Then, by substituting this \( \psi(0) \) into Eq. (20), \( \psi_2(t) \) is obtained as
\[
\psi_2(t) = \left[ \psi_{21}(t) \psi_{22}(t) \cdots \psi_{2N}(t) \right]^T,
\] (23)
\[
\psi_{2i}(t) = 2\omega_i^2 \frac{1 - \cos \omega_i t_f}{\cos \omega_i t_f} q_{ei} + 2\omega_i^2 \frac{1 - \cos \omega_i t_f}{\cos \omega_i t_f} q_{ni},
\] (24)
where \( q_{ei} \) and \( q_{ni} \) is the \( i \)-th element of \( q_e \) and \( q_n \) respectively. From Eq. (15), \( W = \dot{M}^{-1} \), and Eq. (23), the optimal control \( u(t) \) can be computed by
\[
u(t) = -\dot{M}\Phi\psi_2(t).
\] (25)

Furthermore, by Eq. (25) and \( W = \dot{M}^{-1} \), the cost function can be expressed as
\[
J = \frac{1}{2} \int_0^{t_f} u^T(t) M^{-1} u(t) dt = \frac{1}{2} \int_0^{t_f} \psi_2^T(t) \psi_2(t) dt,
\]
\[
= \sum_{i=1}^{N} 2\omega_i^2 \frac{1 - \cos \omega_i t_f}{\cos \omega_i t_f} q_{ei}^2 + \sum_{i=1}^{N} 2\omega_i^2 \frac{1 - \cos \omega_i t_f}{\cos \omega_i t_f} q_{ni}^2,
\] (26)
where we use the fact that \( \Phi^T \dot{M} \Phi = I \). In the above equation, it is easily confirmed that
\[
\frac{2\omega_i^2(1 - \cos \omega_i t_f)}{\omega_i t_f + \sin \omega_i t_f} \geq 0.
\] (27)

Hence it is understood that the optimal spring mounting position that minimizes \( J \) is always \( q_{ni} = 0(i = 1, 2, \cdots, N) \), i.e., \( q_n = 0 \). Therefore in the following, we analyze the problem by setting \( q_n = 0 \).

It should be noted that some of \( \omega_i \) might be zero depending on the structure of the stiffness matrix \( K \). By using the L'Hospital's theorem, we can get
\[
\lim_{\omega_i \to 0} \frac{2\omega_i^2(1 + \cos \omega_i t_f)}{\omega_i t_f - \sin \omega_i t_f} = \frac{24}{t_f^2}.
\] (28)

Hence, the relationship between the minimum value of the energy consumption \( J \) and the operating time \( t_f \) including the case of \( \omega_i = 0 \) can be summarized as follows
\[
J(t_f) = \sum_{i=1}^{N} J_i(t_f),
\] (29)
\[
J_i(t_f) = \begin{cases} 2\omega_i^2 \frac{1 - \cos \omega_i t_f}{\omega_i t_f - \sin \omega_i t_f} q_{ei}^2 & (\text{if } \omega_i \neq 0) \\ \\
\frac{24}{t_f^2} q_{ei}^2 & (\text{if } \omega_i = 0) \\
\end{cases}
\] (30)

Equations (29) and (30) show that the total consumed energy of horizontal manipulators with springs can be reasonably understood as the sum of the consumed energy corresponding to the each mode.

The optimal trajectory in modal coordinates \( q(t) \) is derived by substituting \( \psi(0) \), Eqs. (12) and (13) into Eq. (21) as follows
\[
q_1(t) = q(t) = [q_1(t) \quad q_2(t) \cdots q_N(t)]^T,
\] (31)
\[
q_i(t) = -q_{ei} \cos \omega_i t - q_{ei} \frac{\omega_i t \cos \omega_i t - \sin \omega_i t)(1 + \cos \omega_i t_f)}{\sin \omega_i t_f - \omega_i t_f} - q_{ei} \frac{\omega_i t \sin \omega_i t \sin \omega_i t_f}{\sin \omega_i t_f - \omega_i t_f},
\] (32)

Then, the optimal trajectory in physical coordinates \( \theta(t) \) can be obtained by \( \theta(t) = \theta_m + \Phi q(t) \).
Table 1. Parameters of the 2DOF manipulator.

<table>
<thead>
<tr>
<th>link</th>
<th>i</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>length</td>
<td>$l_i$ [m]</td>
<td>0.250</td>
<td>0.160</td>
</tr>
<tr>
<td>center of mass</td>
<td>$s_i$ [m]</td>
<td>0.125</td>
<td>0.080</td>
</tr>
<tr>
<td>mass</td>
<td>$m_i$ [kg]</td>
<td>14.25</td>
<td>10.00</td>
</tr>
<tr>
<td>moment of inertia</td>
<td>$I_i$ [kgm$^2$]</td>
<td>0.430</td>
<td>0.244</td>
</tr>
</tbody>
</table>

Figure 2. Planar 2DOF manipulator.

2.3 Validation of the obtained optimal solution

The analytical solution obtained in the previous section is based on the linearized equations of motion (8). Hence, we should examine the influence of the nonlinearity ignored in the analysis. Therefore, in this section, we compare the analytical solution with the numerical one considering the full nonlinear dynamics by a general purpose optimal trajectory planning algorithm for multi-body systems [5]. Here we consider the planar 2DOF manipulator shown in Fig. 2 as an example. The parameters used for the simulation are given in Table 1.

The equations of motion of the planar 2DOF manipulator with springs can be expressed as

$$
\begin{bmatrix}
I_1 + I_2 + 2m_2l_1s_2\cos\theta_2 + m_2(l_1^2 + s_2^2) + m_1s_2^2
& \text{sym.} \\
l_2 + m_2(s_2^2 + l_1s_2\cos\theta_2) & I_2 + m_2s_2^2
\end{bmatrix}
\begin{bmatrix}
\dot{\theta}_1 \\
\dot{\theta}_2
\end{bmatrix}
+
\begin{bmatrix}
-m_2l_1s_2\sin\theta_2(2\dot{\theta}_1\dot{\theta}_2 + \dot{\theta}_2^2) \\
m_2l_1s_2\sin\theta_2\dot{\theta}_1^2
\end{bmatrix}
= -
\begin{bmatrix}
k_1 & 0 \\
0 & k_2
\end{bmatrix}
\begin{bmatrix}
\theta_1 - \theta_{n1} \\
\theta_2 - \theta_{n2}
\end{bmatrix}
+ \begin{bmatrix}
u_1 \\
u_2
\end{bmatrix}.
$$

As an example, we consider the motion under the initial and final conditions $\theta_0 = [-30, -30]^T$ deg, $\theta_f = [30, 30]^T$ deg. Fig. 3 shows the comparison between the analytical solution and the numerical one of $J$ for the case without spring, i.e., $k = [k_1, k_2]^T = [0, 0]^T$ Nm/rad. Since the centrifugal and Coriolis forces become larger as the moving speeds become faster, the difference between the two becomes larger as $t_f$ becomes shorter. However, we can confirm that the difference converge to zero as $t_f$ is longer, i.e., the moving speeds become slower. Fig. 4 shows the comparison for the three cases of $k = [k_1, k_2]^T = [30, 0]^T, [0, 30]^T, [30, 30]^T$ Nm/rad. If fairly strong springs are used, the spring forces become predominant over the centrifugal and Coriolis forces. Therefore, we can observe that both results are well coinciding everywhere. Since the analytical solution is well approximating the characteristics of the exact one, it is proposed that the analytical solution obtained in the previous section can be used for analysis and design of planar manipulators with springs.
2.4 Optimal operating time

In this subsection, we derive a condition for an optimal operating time $t_f$. If some $\omega_i$ become zero, an optimal operating time $t_f$ does not exist since $J$ has the terms that decrease monotonously with $t_f$, see Eq. (30). Therefore, in the following, we discuss the case that all $\omega_i$ are non-zero. If we consider $t_f$ as the quantity that may take any value, $H(t_f) = 0$ should be satisfied from the transversality condition. Moreover, since Eq. (16) does not contain $t$ explicitly, \( \partial H / \partial t = 0 \), it holds $H=$const along optimal trajectories. Hence the condition $H(0) = 0$ should be satisfied. By substituting the Eqs. (12) and (13), $\mathbf{q}_n = 0$, and $\psi(0)$ into Eq. (16), one can get

$$H(0) = \psi_2^T(0) \mathbf{q}_e - \frac{1}{2} \psi_2^T(0) \psi_2(0) = \sum_{i=1}^{N} \frac{-2 \omega_i^3 t_f \sin \omega_i t_f}{(\omega_i t_f - \sin \omega_i t_f)^2} q_{ei}^2.$$  

(34)

From Eq. (34), it follows that $H(0) = 0$ is satisfied if $\sin \omega_i t_f = 0$ ($i = 1, 2, \cdots, N$) or equivalently

$$\omega_i t_f = r_i \pi \quad (i = 1, 2, \cdots, N),$$  

(35)

where $r_i$ is an integer. When the condition (35) is satisfied, the second and third term of Eq. (24) vanishes and the optimal trajectory in modal coordinate reduce to harmonic vibrations

$$x_i(t) = q_i(t) = [q_1(t) q_2(t) \cdots q_N(t)]^T,$$  

(36)

$$q_i(t) = -q_{ei} \cos \omega_i t.$$  

(37)

And by substituting Eq. (35) into Eqs. (29) and (30), the cost function can be expressed as

$$J = \sum_{i=1}^{N} J_i = \sum_{i=1}^{N} 2 \omega_i^3 (1 + \cos r_i \pi) \frac{q_{ei}^2}{r_i \pi}.$$  

(38)

From Eq. (38), it is understood that $J_i$ takes the maximum $4 \omega_i^3 q_{ei}^2 / r_i \pi$ if $r_i$ is an even number and vanishes if $r_i$ is an odd number. Hence, $J$ takes the global minimum if all $r_i$ are odd number resulting in the minimum value of zero.

2.5 Optimal design method for springs

In this section, we consider the problem to design the spring stiffnesses $\mathbf{k} = [k_1, k_2, \cdots, k_N]^T$ that make the consumed energy minimum for a specified time $t_f^*$.

Firstly, from Eq. (35), the natural frequencies $\omega_i$ read as

$$\omega_i = r_i \pi / t_f^* \quad (i = 1, 2, \cdots, N),$$  

(39)
where all \( r_i \) should be selected to be odd number so that all \( J_i \) takes the minimum. Moreover, \( r_i \) should satisfy \( r_1 < r_2 < \cdots < r_N \) since we assumed that \( \omega_1 < \omega_2 < \cdots < \omega_N \). The spring stiffnesses \( k = [k_1, k_2, \cdots, k_N]^T \) should be determined as they satisfy the following characteristic equations

\[
\text{det}[K - \omega_i^2 \hat{M}] = 0 \quad (i = 1, 2, \cdots, N),
\]

Let us define the error vector \( e = [e_1, e_2, \cdots, e_N]^T \) where \( e_i = \text{det}[K - \omega_i^2 \hat{M}] \). Then, the problem here becomes to find \( k \) that satisfies

\[
e(k) = 0.
\]

Solving this nonlinear equation, e.g., by Newton-Raphson method, we can obtain the optimal spring stiffnesses \( k \) that minimizes the energy consumption. Especially, for the case of 2DOF manipulator, we can calculate the optimal spring stiffnesses analytically as follows. Since \( N = 2 \), from the characteristic equations \( \text{det}[K - \omega_i^2 \hat{M}] = 0 \) \( (i = 1, 2) \), two equilateral hyperbolas can be obtained

\[
k_i = \frac{c_i^2}{k_1 - a_i} + b_i \quad (i = 1, 2),
\]

where \( a_i = \hat{m}_{11}((r_i \pi / \tau_i)^2, b_i = \hat{m}_{22}((r_i \pi / \tau_i)^2, c_i = \hat{m}_{12}((r_i \pi / \tau_i)^2, \) and \( \hat{m}_{ij} \) is the \( ij \)-th element of matrix \( \hat{M} \). From these two equations, \( k_1 \) can be calculated analytically as follows

\[
k_1 = \frac{A^+ B^- - C^+ C^- \pm \sqrt{(C^+)^2(C^-)^2 - 2A^- B^-(c_i^2 + c_j^2) + (A^-)^2(B^-)^2}}{2B^-},
\]

where \( A^+ = a_2 + a_1, A^- = a_2 - a_1, B^- = b_2 - b_1, C^+ = c_2 + c_1, C^- = c_2 - c_1 \). Then, by substituting \( k_1 \) into Eq. (42), \( k_2 \) can also be obtained.

We can achieve the minimum energy control of planar robot manipulators by adding the springs with optimal stiffnesses to the joint at the optimal mounting positions \( \theta_n \).

### 2.6 Application example

In this section, the proposed optimal design method for springs is demonstrated for the planar 2DOF manipulator shown in Fig. 2. As in the section 2.3, we consider the motion under the initial and the final conditions \( \theta_0 = [-30, -30]^T \text{deg}, \theta_f = [30, 30]^T \text{deg} \). Here we assume that the operating time is specified as \( t_f^* = 1 \text{s} \). Firstly, we select \( r_1 \) in Eq. (39) as \( r_1 = 1, r_2 = 3 \), then the natural frequencies become \( \omega_1 = \pi, \omega_2 = 3\pi \text{ rad/s} \). Let us first set the spring stiffnesses as \( k = [k_1, k_2]^T = [30, 30]^T \text{Nm/ra}d \). Though the consumed energy \( J \) corresponding to this spring stiffnesses is included in Fig. 4, we show it again on a different scale with \( J_1 \) and \( J_2 \) for the first and second mode in Fig. 5. Since the minimum of \( J_1 \) and \( J_2 \) do not coincide, \( J \) is not vanishing. Therefore, we optimize the spring stiffnesses. From Eqs. (42) and (43), the optimal values of spring stiffnesses can be obtained as \( k = [k_1, k_2]^T = [21.854, 14.182]^T \text{Nm/ra}d \). The consumed energy \( J \) corresponding to this spring stiffnesses is shown in Fig. 6 with \( J_1 \) and \( J_2 \). We can observe that the minimum of \( J_1 \) and \( J_2 \) coincides at \( t_f^* = 1 \text{s} \) and therefore \( J \) is vanishing at \( t_f^* = 1 \text{s} \). In Fig. 6, the minimum value of \( J \) corresponding to the case without spring, i.e., a conventional robot manipulator, is also shown. We can confirm that the consumed energy is strongly reduced at the design point \( t_f^* = 1 \text{s} \). This proves the effectiveness of the proposed method.

### 3 Design and Control of Energy Saving Manipulator

The energy saving control method using springs proposed in the chapter 2 utilizes natural modes of vibration of the system. On the other hand, existing robot manipulators have direct drive or geared motors at the joints and therefore free vibrations do not occur. Therefore, the proposed method cannot be directly applied to conventional manipulators. Hence in this chapter, we consider the practical design problem to realize the proposed energy saving manipulator concept. We develop a prototype 2DOF manipulator to validate realizability and effectiveness of the proposed method.
4 Design of energy saving manipulator

In chapter 2, we consider a horizontal manipulator model equipped with rotational springs as shown in Fig. 1 or Fig. 7(a), respectively. However, for the rotational spring it is usually difficult to adjust its stiffness and mounting position, and so we impose in our experimental setup rotational stiffness between neighboring links by using two linear springs and a special spring holder as shown in Fig. 7(b). We denote by $k_{ti}$ the linear spring stiffness, $l_{ti}$ the distance between the joint and spring mounting point on the holder, then the applied torque $T_i$ by springs about a joint can be approximated as

$$T_i = -2k_{ti}l_{ti}^2 \sin \theta_i \cos \theta_i \simeq -2k_{ti}l_{ti}^2 \theta_i \equiv -k_i \theta_i.$$

(44)

From the above equation, it is understood that we can convert linear spring stiffness $k_{ti}$ to corresponding rotational spring stiffness $k_i$ by

$$k_i = 2k_{ti}l_{ti}^2.$$

(45)

Next, let us consider the installation position of actuators. Since the proposed method utilizes the free vibration of the system, all joints must be able to rotate freely. Therefore, we can not install motors at the joints, instead, we introduce controlled reaction wheels at an arbitrary point on the link and add driving torques from them as shown in Fig. 8.

The equations of motion of the links and the reaction wheels in Fig. 8 can be derived as follows

$$M_{\theta \theta} \ddot{\theta} + M_{\theta \phi} \ddot{\phi} + h = -K(\theta - \theta_n),$$

(46)

$$M_{\phi \theta} \ddot{\theta} + M_{\phi \phi} \ddot{\phi} = \tau,$$

(47)

where $\theta = [\theta_1, \theta_2, \ldots, \theta_N]^T$ is the vector of joint variables, $\phi = [\phi_1, \phi_2, \ldots, \phi_N]^T$ is the vector of rotation angles of reaction wheels, $M_{\theta \theta}, M_{\theta \phi}, M_{\phi \theta}, M_{\phi \phi}$ are the inertia matrix, $h$ is the vector of centrifugal and Coriolis forces, $\tau = [\tau_1, \tau_2, \ldots, \tau_N]^T$ is the vector of driving torques of reaction wheels, $K = \text{diag}[k_1, k_2, \ldots, k_N]$ is the stiffness matrix, $\theta_n$ is the vector of spring mounting positions.
By eliminating \( \phi \) from Eqs. (46) and (47), and defining
\[
M = M_{\theta \theta} - M_{\phi \theta}^{T} M_{\phi \phi}^{-1} M_{\phi \theta},
\]
\( u = -M_{\phi \theta}^{T} M_{\phi \phi}^{-1} \tau \),
then the following equations of motion are obtained
\[
M \ddot{\theta} + h = -K(\theta - \theta_n) + u. \tag{48}
\]

Since Eq. (48) has the same form as Eq. (1), the energy saving control method proposed in the chapter 2 can be directly applied to this novel energy saving manipulator systems.

### 4.1 Control of energy saving manipulator

From Eqs. (36) and (37), the minimum energy trajectory between the initial position \( \theta_0 = [\theta_{10}, \theta_{20}, \cdots, \theta_{N0}]^T \) and the final position \( \theta_f = [\theta_{1f}, \theta_{2f}, \cdots, \theta_{Nf}]^T \) can be expressed as
\[
\theta(t) = \theta_m + \Phi q(t)(\equiv \theta_d(t)), \tag{49}
\]
\[
q(t) = [-q_{e1} \cos \omega_1 t, -q_{e2} \cos \omega_2 t, \cdots, -q_{EN} \cos \omega_N t]^T. \tag{50}
\]

Theorically, if we move the links to the position \( \theta_0 \) by applying the external torques for the first time only, then repetitive movement between \( \theta_0 \) and \( \theta_f \) can be achieved continuously without any additional input torque \( u \). However, practically, due to the existence of friction and air resistance, the link motions attenuate gradually with time. Hence, let the desired trajectory \( \theta_d(t) \) be given by Eq. (49), and following feedback controller is introduced
\[
u = M \{ \ddot{\theta}_d - \alpha (\dot{\theta} - \dot{\theta}_d) - \beta (\theta - \theta_d) \} + K(\theta - \theta_n). \tag{51}
\]

By substituting Eq. (51) into the equations of motion (48), it can be seen that the behavior of the error vector \( e(t) \equiv \theta(t) - \theta_d(t) \) is governed by the equation
\[
\dot{e}(t) + \alpha e(t) + \beta e(t) = 0, \tag{52}
\]
where \( \alpha \) and \( \beta \) are constant matrices that guarantee asymptotic stability. Eq. (52) means that \( \theta(t) \) converges to \( \theta_d(t) \), which means that proposed energy saving control method is realized.

### 4.2 A prototype 2DOF manipulator and experimental results

To validate the proposed energy saving control method, and practical design and control approach, we develop a prototype 2DOF manipulator. The developed energy saving manipulator is shown in Figs. 9 and 10. The main specifications of the experimental setup are in Tables 2 to 4. In this experimental device, reaction wheels are driven by DC motors and installed such that their rotational center coincide with the joint axes. The rotary encoders are used to obtain rotational angles of the links. And the electro-magnetic breaks are equipped to the joints so that we can keep holding the links at an arbitrary positions.

As an example, let us consider the repetitive movement between \( \theta_0 = [0,0]^T \) deg and \( \theta_f = [60,60]^T \) deg. The operating time between two points are set as \( t_f = 2 \) s. If we choose \( r_i \) as \( r_1 = 1, r_2 = 3 \), the optimal rotational spring stiffnesses can be obtained as \( k = [k_1, k_2]^T = [0.374, 0.085]^T \) Nm/rad.
Table 2. Parameters of the 2DOF energy saving manipulator.

<table>
<thead>
<tr>
<th>link</th>
<th>( i )</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>length ( l_i ) [m]</td>
<td>0.2500</td>
<td>0.2250</td>
<td></td>
</tr>
<tr>
<td>mass ( m_i ) [kg]</td>
<td>0.8097</td>
<td>0.6747</td>
<td></td>
</tr>
<tr>
<td>moment of inertia ( I_i ) [kgm^2]</td>
<td>0.0062</td>
<td>0.0037</td>
<td></td>
</tr>
</tbody>
</table>

Table 3. Parameters of the reaction wheel.

<table>
<thead>
<tr>
<th>reaction wheel</th>
<th>( i )</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>mass ( m_{wi} ) [kg]</td>
<td>0.0484</td>
<td>0.0484</td>
<td></td>
</tr>
<tr>
<td>moment of inertia ( I_{wi} ) [kgm^2]</td>
<td>( 2.767 \times 10^{-5} )</td>
<td>( 2.767 \times 10^{-5} )</td>
<td></td>
</tr>
</tbody>
</table>

Table 4. Parameters of the motor.

<table>
<thead>
<tr>
<th>motor</th>
<th>( i )</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>mass ( m_{di} ) [kg]</td>
<td>0.0750</td>
<td>0.0750</td>
<td></td>
</tr>
<tr>
<td>moment of inertia ( I_{di} ) [kgm^2]</td>
<td>( 9.250 \times 10^{-6} )</td>
<td>( 9.250 \times 10^{-6} )</td>
<td></td>
</tr>
</tbody>
</table>

Figure 9. Prototype 2DOF manipulator.
Figure 10. Enlarged view of the joint part.

from Eq. (42) and (43). By using Eq. (45), \( k \) is converted to the linear spring stiffnesses as \( k_i = [k_{i1}, k_{i2}]^T = [18.706, 4.237]^T \text{N/m} \). Then we adjust the linear spring stiffnesses of the experimental device as close as possible to the optimal values. After that we conduct trajectory tracking control experiments by using the control law of Eq. (51). Figures 11 and 12 show the results of the trajectory tracking test for the 1st and 2nd joint angles respectively. In these figures, blue line shows the optimal (reference) trajectory calculated by Eq. (49) and (50), and red line shows the actual trajectory measured by encoders. From these figures, it can be seen that the tracking error is fairly small and trajectory tracking is almost achieved. Figures 13 and 14 show the driving torques of 1st and 2nd actuators respectively. In these figures, the red line shows the reaction wheel driving torque of proposed energy saving manipulator and blue line shows the joint driving torque required by the conventional manipulator to perform the same task. Let us compare the energy consumption of the proposed and the conventional manipulators by the following cost function

\[
J(t) = \frac{1}{2} \sum_{i=1}^{2} \int_0^t u_i^2(\tilde{t})d\tilde{t}.
\]  

Fig. 15 shows the result. The value of \( J(t_f) \) for the proposed manipulator was 10.57, in contrast for the conventional manipulator was 173.5. This result proves that the proposed concept is effective and it can reduce the energy consumption.
5 CONCLUSIONS
In this study, we considered a method for reducing energy consumption of planar robot manipulators by adding springs to the joints and utilizes the potential energy effectively. Firstly, we proposed a simultaneous optimization method for springs and trajectories based on the optimal control theory. Then, we discussed the practical design and control problems to realize the proposed energy saving manipulator concept. Finally, we developed a 2DOF energy saving manipulator prototype and show the effectiveness of the proposed method through experiments.

ACKNOWLEDGMENT
This research was supported by KAKENHI, Grants-in-Aid for Young Scientists (B)(No. 26820076).

REFERENCES
Figure 15. Comparison of energy consumption.


Dynamic Analysis using Numerical Multi-body Approach for Quadruped Robots

Mariapaola D’Imperio*, Ferdinando Cannella*, Fei Chen*,
Luca Carbonari*, Daniele Catelani# and Darwin G. Caldwell*

* Istituto Italiano di Tecnologia
Via Morego 30-16163, Genova, Italy
{mariapaola.dimperio, ferdinando.cannella, fei.chen, luca.carbonari, darwin.caldwell}@iit.it.
# MSC.Software srl
Via Santa Teresa 12, Torino, Italy
daniele.catelani@mscsoftware.com

ABSTRACT
Nowadays the panorama of quadruped robots widely ranges from from robust to agile applications, from electrically to hydraulically actuated. They are capable of performing complex dynamic tasks like trotting, galloping and high speed running on uneven terrains. Dealing with these skills is a challenging issue because of the need of taking into account the reciprocal effects between mechanics and control. Traditionally such mutual interaction is developed by means of two semi-independent models and the two aspects get in touch mostly during the physical experimental test, when it is difficult to understand whether the troubles come from one of the two sides or from their integration. This work aims at introducing an integrated approach to get rid of such obstacles by means of a cooperation between simulated multi-body mechanical systems and control algorithms. In particular, details are provided about the modelling strategies adopted to assess the challenging case of a hydraulically actuated walking machine.

Keywords: Dynamics, Hydraulics, Co-simulation, Multi-legged system.

1 INTRODUCTION
Robots achieving truly dynamic motion tasks within their environment represent a challenging emergent frontier of robotics. The successful design of platforms capable of attaining desired performances in terms of mobility and controllability is guaranteed through an optimum combination of several engineering fields, namely mechanics, control and electronics. Efficient integration of these methodologies is hard and exacerbated by extreme dynamic tasks that robots are expected to perform. Simulation tools represent a rising solution to such issues, since they are aimed at reproducing systems behaviour for the purposes of design optimization and controller development [23].

In this paper a case-study model of a legged robot is developed, based on the exploitation of software environments, dedicated at both Multi-Body Model (MBM) and control design, and above all on their cooperation. Such co-simulation allows a contemporaneous study of the dual aspects of mechatronic systems, i.e. mechanics and control.

The last frontier of legged robots is represented by quadrupeds which perform complex dynamic tasks like trotting, galloping and high speed running. KOLT [22], BigDog [19], HyQ [25], LS3 [10] and Wildcat are practical examples for whose trajectory generation has to be dealt together with dynamic and equilibrium stability [21]. Their capabilities pushed the limits of mechanical control and design with their peculiar inconveniences such as the impulsive Ground Reaction Forces (GRF) that arise during the motions.

By tradition, the approach adopted to deal with these requirements is based on the trade-off between active and passive compliance [8]. The active compliance is a strategy consisting of tuning the robot stiffness and damping properties to make it absorb the energy due to the GRF [12]. This solution does not achieve all the requested dynamic performances while the control system becomes complex due to bandwidth limitations. On the other hand, the passive compliance, is a
solution obtained by acting on the mechanical properties of the machine [20] such as adding to the structure springs or shock absorbers in order to preserve it from the damage. Nevertheless, this solution reduces the joints controllability due to the slow system response. It also adds additional mass, increases the impact forces magnitude and modifies system natural frequencies, affecting controller efficiency at certain speeds. For these reasons, researches look for a tradeoff between the two solutions.

In both of the aforementioned methods, the mechanical and the control designers work for an identical target and create two different models for solving two aspects of the same problems. The mechanical designer develops models with CAD software and computes the bodies geometric properties. Afterward, the rigid body inertia properties are used for creating the Multi-Body Model (MBM) by using software like Gazebo, V-Rep, Webots, SL, Robotran, etc [15]. This current method, pictured in Fig.1-(a), permits both the offline simulation and the model based control of the physical robot. Besides being very computationally efficient, this approach also allows a quick validation of the MBM through experimental results. However, the use of rigid bodies and the non-interactive connection with the CAD do not permit to investigate the robot structural behavior (flexibility, stress, strain, material compliance and non-linearity, durability, fatigue, etc.) that affects the control feedback and then its stability [11]. Furthermore this mutual interaction depends on the overall force increments; that means, the more the robot is stressed, the more the performance are limited [27]. Tackling the design in such way makes difficult to improve rapidly the design and/or to realize new versions of the robot. Thus, it looks reasonable to look for a

Figure 1. Design process for legged robots. (a) Traditional approach: no mutual correlation between the mechanical and control model. (b) Proposed approach: single model, based on the Virtual Prototyping and cosimulation between mechanical and control model.
strategy to reduce the time of the validation process loop.

The method here proposed, based on Virtual Prototyping (VP) techniques, may be able to overpass this issue. Figure 1-(b) shows the concept: at the first stage the CAD model and the VP share the model geometrical and mechanical properties; then the connections between VP and the control allows simulating the robot performances very accurately thanks to the a priori identification of the model parameters [7]; at last, when the best design is chosen and the whole physical prototype is built and assembled, the VP model results can be compared with experimental tests; so the model becomes more reliable and it can be safely used for further improvement of the robot.

Hence, the aforementioned method is suitable for investigating the active-passive compliance tradeoff exploiting the cosimulation between structural dynamics and control. The combined effects of a control system on a MBM can be tested and verified thanks to a co-simulated VP model. Thanks to this method it is possible to reduce the risk of controlling mechanical systems with fallacious control laws before testing the algorithms on the actual prototype.

The main advantage in using such a model is the possibility of testing the robot under extreme performance conditions, while considering reductions of computational burden in simulation and control application. It is the case of impact dynamics for whose it is difficult to have an effective force control since the contact time is too short. Even if the model will not be able to overpass this problems for control tasks, it will for sure be able to predict such structural response in simulation. Every improvement, both from the structural and from the control side, can be immediately simulated within a global analysis, with positive effects on the whole project being the time for one simulation shorter than the time needed for machining a modified prototype or for arrange an experimental test. At last, since it is a virtual model, it is possible to run different simulations at the same time. The higher number of the design versions that it is possible to investigate thanks to this method will yield the improvement of the whole concept quality [1].

Nevertheless, the dynamic modelling of a system has some drawbacks due to parameters uncertainties, coming from measurements complexity or variations from real system data, change of payloads during runtime operations and environmental disturbances like collision that can cause deviations from ideal or predicted movements or behaviours. Thus, having a valid MBM requires an identification of all the parameters involved especially when complex systems are involved such as hydraulic actuators. For these reasons, a preliminary identification of the aforementioned parameters has been performed and its results and are introduced and discussed in this manuscript.

The remainder of the paper is organized as follows. Section II contains the robotic platform description, Section III addresses the numerical modelling design, Section IV describe experimental tests and their results while Section V addresses discussion and conclusions.

2 ROBOTIC LEG SYSTEM: DESCRIPTION

The robotic platform presented in this manuscript is a quadruped robot called HyQ (Hydraulically Quadruped, see Fig.2-(a) [25]. The robot weights 70 [kg], it is 1 [m] long and it is mainly composed by a torso and four legs. The torso doesn’t influence the system mobility, it is designed for carrying the instruments and supporting the four legs. The legs, instead, have two degrees of freedom in the sagittal plane, namely the hip and knee flexion/extension (−40° ≤ q1 ≤ 60° and −140° ≤ q2 ≤ −20° respectively).

HyQ is designed to achieve a wide range of dynamic tasks: from walking to trotting, from jumping to climbing. These dynamic capabilities are challenging to deal with and achieving satisfactory results requires a perfect integration between structural aspects and control. For these reasons, HyQ represents the perfect test bench for the multi-environmental simulated approach object of this study. Due to the several physical natures of the involved phenomena, this approach turns particularly useful since it allows to take into consideration, at a same time, hydraulics as well as rigid body dynamics. The work here presented focuses mainly on the modelling of one of the
robot legs and represents a preliminary step to product of a complete MBM of the HyQ.

2.1 Dynamics

The robotic leg dynamics can be formulated by the virtual work principle, see Equation (1) [4, 5],

$$\delta q^T \tau + \sum_{i=1}^{7} \delta x_i^T W_i = 0$$  \hspace{1cm} (1)

that requires the definition of both inertial forces and actuation torques for each body, so as:

$$W_i = \begin{bmatrix} n_i \\ F_i \end{bmatrix} = \begin{bmatrix} -0^0 \Gamma_0^0 \omega_k - 0^0 \omega_i \wedge 0^0 \Gamma_0^0 \omega_i + n_{e,i} \\ m_i (g - \ddot{x}_i) + f_{e,i} \end{bmatrix}$$  \hspace{1cm} (2)

After several manipulations [13] the Equation (1) can be expressed in the canonical form:

$$\tau + F_e = M(q) \ddot{q} + V(q, \dot{q}) + G(q)$$  \hspace{1cm} (3)

This last one is written in the state space and it links the joints actuation to the leg kinematics. A detail of the terms of the Equations (1)-(2) and (3) is given in the Appendix.

2.2 Actuation system

The robotic leg actuation system is composed of two hydraulic asymmetric cylinders (Fig.3-(a)). Each one of them is composed by a hollow cylindrical body and a piston, which is linked to the leg mechanical components. The asymmetry is due to the presence of two different chambers area in the piston: the big one is called piston area $A_p$, the small one instead is an annular area and it is defined road area. The force exerted by this kind of actuation depends on the difference of pressure between the two chambers ($p_a$ and $p_b$ respectively):

$$F = A_p p_a - \alpha A_p p_b$$  \hspace{1cm} (4)

where $\alpha$ represents the ratio between the piston area and the road one.
The chamber pressure formula derives from the mass conservation principle, that states that "for any system closed to all transfers of matter and energy, the mass of the system must remain constant over time", that can be expressed, in ideal conditions, as:

\[ q_{in} - q_{out} = \frac{d(\rho V)}{dt} = \frac{dp}{dt} V + \frac{dV}{dt} \rho \tag{5} \]

where \( \frac{dp}{dt} = \frac{\rho}{\beta} \frac{dP}{dt} \) and \( \beta \) is the bulk modulus, characterizes the fluid compressibility.

Thus in chamber \( a \), assuming that there is no internal leakage and that \( q_{out} = 0, V = v_a \), the formula for the relative pressure is described in the Equation (6)

\[ \dot{p}_a = \frac{\beta e}{v_a} (q_a - A_p x_p) \tag{6} \]

On the other hand in chamber \( b \), considering that there is only an exit flow without any add of mass, the Equation (5) becomes

\[ \dot{p}_b = \frac{\beta e}{v_b} (-q_b - \alpha A_p x_p) \tag{7} \]
where \( q_a \) and \( q_b \) represent the chambers flows, \( v_a \) and \( v_b \) are the chambers volumes and \( x_p \) is the stroke position.

The transmission of the fluid from the pump to the actuators, in the studied case, is regulated by servovalves. The fluid motion is a common flow through an orifice that depends not only from the dimension and characteristic of the orifice, but also from the difference of pressure across it:

\[
q = C_d \alpha \sqrt{\frac{2 \Delta p}{\rho}} = C_d W u_v \sqrt{\frac{2 \Delta p}{\rho}} = C_d W \frac{\sqrt{2}}{\sqrt{\rho}} u_v \sqrt{\Delta p} = K_v u_v \sqrt{\Delta p}
\]

where \( C_d \) is the discharge coefficient, \( \rho \) represents the oil density, \( K_v \) is the valve gain while \( u_v \) represents the input current.

In chamber \( a \), the flow is described by:

\[
q_a = \begin{cases} 
K_v u_v \sqrt{p_a - p_a}, & u_v > 0 \\
K_v u_v \sqrt{p_a - p_t}, & u_v \leq 0
\end{cases}
\]

while in chamber \( b \), the flow is:

\[
q_b = \begin{cases} 
K_v u_v \sqrt{p_b - p_t}, & u_v > 0 \\
K_v u_v \sqrt{p_s - p_b}, & u_v \leq 0
\end{cases}
\]

where \( p_s \) is the pressure derived form the pump and \( p_t \) is the pressure to the reservoir.

Summarizing the Equations from (5) to (10) in Equation (4) it possible to get the formula that describes the derivatives of the force of an hydraulic asymmetric cylinder, see Equation (11).

\[
\dot{F}_h = \begin{cases} 
- \beta e A_p^2 \left( \frac{1}{v_a} + \frac{\alpha^2}{v_b} \right) x_p^2 + \beta e A_p K_v \left( \frac{\sqrt{p_s - p_a}}{v_a} + \frac{\alpha \sqrt{p_b - p_t}}{v_b} \right), & u_v > 0 \\
- \beta e A_p^2 \left( \frac{1}{v_a} + \frac{\alpha^2}{v_b} \right) x_p^2 + \beta e A_p K_v \left( \frac{\sqrt{p_a - p_t}}{v_a} + \frac{\alpha \sqrt{p_s - p_b}}{v_b} \right), & u_v \leq 0
\end{cases}
\]

2.3 Control

The control algorithm for a robotic quadruped is usually developed according to its bioinspired nature. Animals usually choose their gaits and speed following the minimum energy consumption principle [14]. For these reasons the control architecture of the HyQ leg is developed according to the desired dynamic performance requested [3].

Literature presents different approaches to face with the non-linearity and locomotion of such robotic systems. Jacazio and Bassolini demonstrated in [16] that a PID controller provides satisfactory results, both in low speed and in high speed conditions. Due to that, even for HyQ robot, the low level control is a PID [3]. More in details it estimates the error between the actual position and the desired position for each joint with the aim of estimating the current proper value in input for the servovalves that govern the actuation system, (Fig.3-(b)). This low level control loop aims at governing the position of each point according to the set point managed by an external high level control, in order to perform the equilibrium balancing.

3 ROBOTIC LEG SYSTEM: NUMERICAL MODEL

The Leg Numerical Model (LNM) (Fig. 3-(c)) is developed by MSC Adams and Simulink software respectively and it is aimed to deal with all the dynamic issues of the multi-body model. That model contains several elements, as the following:
• **System.** The LNM system is composed by five main assemblies: Slider, Upperleg, Lowerleg, Hip piston and Knee piston. Each of them has been modelled as a rigid body with distributed inertia. The LVM internal constraints are represented by *cylindrical* and *translational* connections while its external one is *translational*.

• **Word interaction.** The interaction with the external environment is represented by a contact law. It is established both by a normal and a tangential force. The normal force (Vertical magenta arrow in Fig.1-(b)) one has been modelled according to the Hertzian law (12)

\[ F_n = k_c \delta^n + c_c \dot{\delta} \]  \hspace{1cm} (12)

where \( k_c \) is the stiffness, \( \delta \) represent the penetration depth, \( n \) is an exponent and \( c_c \) is the damping of contact force. The tangential force (Horizontal blue arrow in Fig.1-(b)) has been modelled by a velocity-based on friction model (13)

\[ F_t = \begin{cases} \mu_s W & \\ \mu_d W & \end{cases} \]  \hspace{1cm} (13)

According to the relative speed between bodies, this model determines the value of that force by applying a proper coefficient (\( \mu_s \) for the static condition and \( \mu_d \) for the dynamic one) to the value of the weight of the structure \( W \). All these parameters depend on the nature of the bodies that collide. In the work presented here the empirical values for the aforementioned parameters are \( k_c = 2855 \text{Nmm}^{-1} \), \( \delta = 0.1 \text{mm} \), \( n = 1.1 \), \( c_c = 1 \text{Nmm}^{-1} \text{s}^{-1} \), \( \mu_s = 0.7 \), \( \mu_d = 0.55 \).

• **Sensors.** Sensors are reproduced by angle, position, velocity and acceleration *measurements* in the LNM.

• **Actuators.** The LNM actuation system is modelled with a continuous state function by means of a S-function aimed to estimate the hydraulic force variation exerted from the pistons, receiving in input the control signal coming from the upper level control, [3]. The actuation force is then integrated involving the structural dynamical effects by means of the co-simulation between the control and the mechanical system.

The \( \dot{F}_h \), as declaimed in Equation (11), depends not only from the control input but also from other parameters and variables. Most of them depend on the system dynamics and geometry, except two, namely the \( \beta \) and the \( K_v \). Their value can be taken from datasheets if it is not possible to identify them by a proper procedure. In this work, an identification process for those parameters is carried out. It is based on a Least Square Method and it requires data coming from an extended campaign of experimental tests.

• **Control** At each simulation step \( q_1 \) and \( q_2 \) actual position (Fig.3-(b)) are measured in MSC Adams and forwarded to SIMULINK. These values represent the input variables of the control system. Afterwards, the actuation algorithm estimates the error between the actual position and the desired position and send back to MSC Adams the torque proper value to apply for the next step.

4 **EXPERIMENTAL TESTS**

The experimental setup of the leg (Fig.3-(c)) consist of one load cell for each actuators strokes(*Burster 8417*, force range 0-5 kN, accuracy \( \pm 0.5 \% \)); one angular encoder for each DOF(*Avago AEDA3300 BE1*, up to 80000 counts per revolution, resolution 0.0045deg) and a displacement sensor mounted on the slider (*Absolute Encoder austriamicrosystem AS5045, signal 12Bit, resolution 0.0879deg*).

The system internal cylindrical connections are represented by bushings while the pistons serve as internal translational connections. A sleeve, moving on a vertical bar and attached to the slider.
guarantees the connection between the leg and the external environment. The leg behaviour, as stated before, is governed by a control system: it has the aim to modify the stiffness and the damping of the whole hydraulic system by adjusting the $P$ and $D$ gain values.

In order to investigate the behaviour of each single piston, during the experimental tests the actuation for the two DOF was decoupled. A series of sinusoidal movement were given in input with different amplitudes and frequencies. The measured quantities were the forces exerted from each piston, the supply and reservoir pressures and the joints rotations Fig.4 -(a).

4.1 Discussion of results

The experimental test data were used in the identification process and the results are resumed in Fig.4 -(b). It is noticed that those are not stable and consistent with the reference datasheets values. Those results are affected by several issues later detailed. Such problems make almost impossible to draw a mathematically based model able to reproduce the actual system dynamics. This is due
to the non-linearities which affect the test-rig and which are in most of the cases nonpredictable. The aforementioned issues can be summarized in:

- presence of the air bubble in the oil: it induces the variation of the bulk modulus magnitude of several percentage points, from 50% in low pressure conditions to 1 – 2% in high pressure conditions, [14];
- pressures signals not constant: most of the theoretical methods require a steady pressure while in the setup we are using it is not possible to have that specific signal (Fig.4 -(a)) [17];
- low sampling frequency: in order to detect a high frequency behaviour, proper of an hydraulic system, it is needed a sampling rate higher than 1[kHz], [17].

For these reasons a different modelling procedure is needed, instead of working with a theory based on modelling it is reasonable to move to an approach based on the system properties as in a black box one.

5 CONCLUSIONS AND FUTURE WORKS

An alternative method of developing a numerical model is presented here for the co-simulation of control and mechanical performance of a quadrupedal robot leg under dynamic operation. The approach presented in this paper can be extended and applied to other robotic system, which have to perform complex dynamic tasks, i.e. humanoids or multi legged robot in general.

Future investigations will be both in the identification process direction and in the one of the co-simulation integration. For the former, a model based identification process is required. Different procedures as Maxwell-Model and NARMAX neural networks are investigating by the authors. For the latter, the next step will be the integration of the full robot MBM both with the low and upper level control architecture. The authors have already developed the MBM for a quadruped robot; at this stage it doesn’t have any balancing control and it able to perform only a squat movement as shown in Fig.4 -(c-d).

Developments in this area will result in the increasing the ability of quadruped robots to perform higher dynamic task to close the gap between the robots and the fast animal who inspired them.

Appendix

- $\delta q$ represents the actuated joints finite virtual displacements;
- $\tau$ is the vector of the applied torques;
- $\delta x$ represents the finite virtual rotation/displacement of the $i^{th}$ body;
- $0 \Psi_i$ represents the inertia tensor of the $i^{th}$ body expressed with respect to the fixed reference frame $\{0\}$;
- $m_i$ is the mass of the $i^{th}$ body;
- $n_{r,i}$ and $f_{r,i}$ are two external actions acting the body;
- $g$ is the gravity acceleration vector $g = \begin{bmatrix} 0 & -9.81 & 0 \end{bmatrix}^T m/s^2$;
- $W_i$ is the wrench acting on the $i^{th}$ body;
- $M(q)$ represents the mechanical system mass matrix:

$$
M(q) = -\Psi^T \left( \sum_{i=1}^{n} J_i^T m_i J_i \right) \Psi
$$
\[ V(q, \dot{q}) = -\Psi^T \left( \sum_{i=1}^{7} J_i^T (m_i \dot{J}_i + \Gamma_i J_i) \Psi + m_i J_i \Psi \right) \dot{q} \]

- \( V(q, \dot{q}) \) includes all the Coriolis and centripetal accelerations:

\[ G(q) \]

- \( G(q) \) groups the effects of gravity acceleration:

\[ F_e \]

- \( F_e \) contains all the external forces and moments:

REFERENCES


ABSTRACT

The paper deals with the multibody dynamic model of the system robot–tool to work with a compliant environment. Manipulator with the PUMA-like kinematics is used, as a model of the robot. The tool has a translational degree of freedom and is firmly fixed to grab of the robot, so its degree of freedom can be treated as a fictitious joint. This system will be used for regenerative medicine, such as massage. Particular attention is paid to the safety performance of procedures when working with a patient, provided by inclusion in the control loop model of the robot. The purpose of this paper is to construct a model, that has characteristics close to real. And further, to build a model of a hybrid position-force control system to work with visco-elastic surface and study the following parameters: the deviation of the tool axis from the surface normal, the deviation of effort when passing through the nodal point of the trajectory.

Keywords: robotics, PUMA-like manipulator, hybrid position/force control, rehabilitation, MATLAB Simulink, Universal Mechanism (UM).

1 INTRODUCTION

This paper considers the model of the manipulator with the PUMA–like kinematics [1]–[2], which is used in mechanical therapy (Fig. 1), specifically massage person [3, 4]. The idea of using a robot for mechanical therapy is not new [5]–[8]. Robotic devices are actively introduced in medicine and everyday life for passive and active development of limbs in the rehabilitation of patients with injuries of the musculoskeletal system, as well as robots to massage the chest, face, head and other parts of the body. While working with the patient great attention are focused on the...
safety of procedures, which is provided by including model of the robot into the control circuit: from «doctor −> robot −> patient» to «doctor −> model of the robot −> robot −> patient». Control algorithms can be practiced on that model.

The purpose of this paper is to construct a model having characteristics close to real and to study the impact of deviation efforts while passing through the nodal point of the trajectory on the effectiveness of mechanotherapy.

2 DYNAMIC MODEL
Description of tree-structure model and appearance of manipulator assembled using UM [9, 10] are shown in Figure 1. The robot consists of seven bodies, submitted by seven graphic images. The images and bodies of working tool and surface, which contacts with manipulator tool, are included in addition to graphic images, which describes the manipulator. To describe the mechanical system, the following model used (1)–(5):

2.1 DC motor mathematical model
\[
\begin{align*}
\mathbf{u} &= iR + C_\omega \dot{\theta}, \\
M &= C_m i,
\end{align*}
\]
where \(i\) – armature current, \(C_m\) – torque constant, \(C_\omega\) – speed constant, \(R\) – armature resistance, \(M\) – motor torque, \(u\) – voltage, \(\dot{\theta}\) – an angular velocity of rotor.

2.2 Full mechanical system model
System of dynamic equation to describe the manipulator with drives:
\[
\begin{align*}
\mathbf{A}(\mathbf{\theta}) \dot{\mathbf{\theta}} + \mathbf{B}(\mathbf{\dot{\theta}}, \mathbf{\theta}) + \mathbf{C}(\mathbf{\theta}) = \mathbf{M} - \mathbf{Q}_e, \\
\mathbf{M} = \mathbf{b}_1 \mathbf{u} - \mathbf{b}_2 \dot{\mathbf{\theta}},
\end{align*}
\]
\(u\) – vector of control actions at the entrance to the drive system.

PD controller used in this problem for generating control signals in order to study the accuracy and quality of transients: \(u_i = -k_1 \Delta \dot{\vartheta} - k_2 \Delta \dot{\vartheta}\), where \(\Delta \dot{\vartheta} = \dot{\vartheta}_i - \dot{\vartheta}_{ipr}\) – mismatch of program and actual trajectories, \(i = 1, \ldots, 6\).

2.3 Contact model
Contact force element «sphere – Z-surface» is used in the UM for the simulation of contact interaction. Contact point is determined by minimizing the distance between the surfaces (e.g., between the sphere and the Z-surface). It uses the simplest model of contact forces: the normal force is determined in accordance with the model of visco-elastic interaction, the tangential force is calculated in accordance with the model of dry friction:
\[
\begin{align*}
\mathbf{N} &= -(c \Delta z + k \Delta \dot{z}) \mathbf{v}, \\
\mathbf{F} &= \begin{cases} 
-f N v/|v|, & |v| > v_f, \\
-f N v/v_f, & |v| \leq v_f,
\end{cases}
\end{align*}
\]
\(N\) – normal reaction force, \(F\) – dry friction force, \(v\) – surface normal, \(\Delta z\) – immersion depth of work tool, \(c, k\) – surface parameters, \(v\) – slip velocity, \(f\) – friction constant, \(v_f\) — sufficiently small.
2.4 Tool force control

Force control law of the tool with a translational degree of freedom [11]:

\[ f = m \left( \frac{k_p}{k_v} e_f - k_v f_t \right) + f_e. \]  

(5)

Here \( e_f = f_d - f_e \) – the difference between the desired force \( f_d \) and measured force \( f_e \), \( k_p, k_v \) – PD controller gains.

We can attain any desired behavior from feedback control system by varying the controller gains in equation (5). We are interested in the case of critical damping when the response of the system is as fast as possible and not oscillate.

3 CONTROL SYSTEM MODEL

Combined position/force control [2], [5], [11] of the system «manipulator – tool» was built in this task to implement the necessary contact interaction. Position control was built for manipulator, force control for tool. The tool has a translational degree of freedom and is firmly fixed to the robot grab, so it is degree of freedom can be seen as a fictitious robot joint.

Model of the robot and the working surface is created in the software package "Universal Mechanism", control model – in MATLAB Simulink (Fig. 2).

Figure 2. UM – MATLAB Simulink.

Figure 3 shows a block diagram of a tool force control system, which has been implemented in the UM.

There are input: tool weight, stiffness coefficient, the gain of the PD controller, velocity of the tool, which is directed along the normal to the contact surface, the desired and the actual force of normal contact, and the output is a driving force, which must be implemented in a translational tool joint.
4 EXPERIMENTS
The set of numerical experiments: the working tool moves along a trajectory, consisting of line segments, the axis of symmetry of the tool is directed along the normal to the surface. 7 nodal points of the trajectory is selected on the work surface, which is a model of the human torso. Thus nodes immersed in the working surface at \( h = 5 \) mm. Parameters of contact interaction:

\[
c = 10000 \text{ N/m}, \quad k = 100 \text{ Ns/m}, \quad f = 0.25 \text{ m}.
\]

Drives parameters:

\[
b_{1i} = 4000, \quad b_{2i} = 1000, i = 1, \ldots, 6.
\]

\[
k_{11} = 100, \quad k_{12} = 0.2, \quad k_{21} = 100, \quad k_{22} = 0.17, \quad k_{31} = 100, \quad k_{32} = 0.14,
\]

\[
k_{41} = 90, \quad k_{42} = 0.11, \quad k_{51} = 90, \quad k_{52} = 0.08, \quad k_{61} = 90, \quad k_{62} = 0.05.
\]

The desired contact force – 20 N. The gains of the force control calculated from numerical experiments. The following Figures 4–5 shows plots of robot and tools joints coordinates, that implement the desired trajectory in the Cartesian space.
The Figure 6 shows a plot of the normal contact force, that occurs between the tool and the work surface for different values of the gains of the force control. As can be seen from the plot 6 (red) for the hybrid position/force control experimentally possible to choose controller gains that the deviation from the desired contact force will not exceed 0.5N.

5 CONCLUSIONS

Path planning algorithm is constructed for linear motion trajectories in Cartesian space. Law between the number of nodal points of the trajectory and force error is constructed. Coefficients of PD-controller was so selected, that the force error did not exceed 2.5% of the given force by the hybrid position/force control.

Conducted computing experiments allow to make following conclusions. Models of interaction of bodies and structural components of the mechanical system, introduced in this paper can be used for efficient modeling robots for mechanotherapy, also software packages for solid modeling may be used. Such models are significantly more efficient in terms of computing and provide sufficiently rich and adequate description of this problem.
REFERENCES


Dynamic Parameter Identification of a Robot in a Simulated Environment

Abdullah Aamir Hayat, Vishal Abhishek, Subir K Saha

Department of Mechanical Engineering,
Indian Institute of Technology Delhi,
Hauz Khas, New Delhi-110016, India
vishalabhishek1691@gmail.com, aamir7hayat@gmail.com, saha@mech.iitd.ac.in

ABSTRACT

Off-line programming of a robot relies on the kinematic and dynamic model of the robot in the simulated environment. This paper discusses the identification of dynamic parameters of a robot in a simulated environment. The input parameters such as torque and joint values are taken from the RoboAnalyzer (RA), a 3D model based robotics learning software. Serial manipulator KUKA KR5 model having six DOF is taken for identification. The reduced dynamic formulation was done using Newton Euler formulation for serial systems. Quintic trajectory is utilized for identification process as it results in exciting all the dynamic parameters of the reduced model. Base parameters for six DOF robot KUKA KR5 was found in simulated environment. The identified model exactly matches with the dynamic model of RA, as measurement noises are absent. The simulated environment of RoboAnalyzer has an advantage of acquiring the joint values and torque data using Inverse dynamics.

Keywords: Dynamic Identification, Simulation, RoboAnalyzer.

1 INTRODUCTION

The standard robot identification procedure consists of modeling, experiment design, data acquisition, signal processing, parameter estimation, and model validation [1]. Lagrangian or Newton Euler methods are mostly used to obtain a mathematical model of the robotic system and are discussed in [2, 3]. The minimal inertial parameters of the robot are then obtained from mathematical model which consists of position, velocity, acceleration, inertial and torque values [4, 5]. The dynamic parameters for a robot manipulator are typically classified as unidentifiable, uniquely identifiable and identifiable in linear combination. The selection of excitation trajectories plays a vital role in number of identified dynamic parameters in reduced model [6].

A robot is in general expensive. Hence its simulation is useful for education, design, testing, research purposes, etc. Nowadays, off-line programming of a robot also relies on kinematic and dynamic models of the robot in the simulated environment. Identifying the robot parameters in the simulated environment is also essential. The dynamic parameter identified in simulated environment results in estimating the joint torques, control and to predict the performance of the designed robot controller in advance. Dynamic parameters of the cylindrical robot in the simulated environment were identified in [7]. In simulation experimental design, data acquisition and signal processing becomes an easier task in comparison to the experimental identification process. Hence, one can test the performance of the dynamic formulation and identification technique without paying attention to those factors.

In this paper RoboAnalyzer [8, 9] a 3D model based robotics learning software capable of performing kinematics and dynamics of a wide range of robots is used for dynamic identification. The contribution and approach of this paper are listed as follows:

- Dynamics of the manipulator is formulated using Newton Euler method [3].
- The dynamic equation obtained is nonlinear in joint parameters is then transformed into equivalent linear form (in constant dynamic parameters) of the robot model.
The industrial robot KUKAKR5 model in RoboAnalyzer (RA) is utilized for getting the joint and torque values with inverse dynamics by applying quintic trajectories with different initial conditions. The base parameters were then found using numerical input of joint values and torques to the linear model.

The results were then verified for a general trajectory provided to these joints with the identified dynamic model and obtained torque data from the RoboAnalyzer software.

This paper is divided into four sections. In Section 2, the dynamic modeling and the base parameters are explained. In Section 3, the data were acquired form the RoboAnalyzer for finding the reduced dynamic model. Section 4 concludes the paper.

2 FORMULATION

In this section the description of the dynamic formulation and the linerized regression model is discussed.

2.1 Dynamic formulation

The robotic manipulator is defined by its kinematic and dynamic parameters of each link. Generally the kinematic description of the robot is done using Denavit-Hartenberg (DH) notation. Among the several variants of the DH parameters the distal DH notation as explained in [3] as modified DH parameters is taken into account. The dynamic parameters, i.e., six moments of inertia terms, three mass moments terms and one mass of link i at the origin of frame i is expressed in Equation (1) appears in the joint torque expression. These are also known as standard inertial parameters (SIP). The definition of DH parameters and the coordinate system assigned for describing the dynamic parameters are explained in Appendix 1.

\[
\begin{bmatrix}
I_x & I_y & I_z & m_{r_x} & m_{r_y} & m_{r_z} & m_i
\end{bmatrix}^T
\]  

(1)

The dynamic formulation chosen here to get equations of motion is Newton-Euler (NE) method [3]. The NE equations are given as,

\[
f_i = m_i \ddot{\theta}_i
\]  

(2a)

\[
n_i = I_i \ddot{\omega}_i + \omega_i \times I_i \omega_i
\]  

(2b)

where, \(f_i\) is the net force acting on link i, \(\ddot{\theta}_i\) is the acceleration of center of mass of link i, \(m_i\) its mass, \(n_i\) net moment about its center of mass, \(I_i\) inertia tensor about center of mass and \(\omega_i\) is its angular velocity. Force and moment balance about center of mass of each link can be used to find joint torques as,

\[
f_{i-1,i} = f_i + f_{i,j+1} + m_i g
\]  

(3a)

\[
n_{i-1,i} = n_i + n_{j+1} + d_i \times f_{i-1,j} + r_i \times f_{i,j+1}
\]  

(3b)

\[
\tau_i = e_i^T n_{i-1,j}
\]  

(3c)

where, \(f_{i-1,j}\) is the force exerted by link i-1, on link i, \(f_{i,j+1}\) is the force exerted by link i on link i+1, \(g\) is the acceleration due to gravity, \(d_i\) is the vector from \(O_{i-1}\) to center of mass of link i, \(r_i\) is the vector from center of mass of link i to \(O_{i+1}\), \(e_i\) is unit vector along \(Z_i\) and \(\tau_i\) is joint i torque.

2.2 Reduced Dynamics or Regressor Form

The mathematical model in Equations (2, 3) were modified to linear parameterized form [2] in standard inertial parameters (SIP) as,

\[
\boldsymbol{\tau} = Y_i(\theta, \dot{\theta}, \ddot{\theta}) \chi_i = Y_i \chi_i
\]  

(4)

\(Y_i(\theta, \dot{\theta}, \ddot{\theta})\) is (number of links \(n\)×number of SIP) regressor matrix. The above equation contains linearly dependent columns which are regrouped by means of numerical method using
QR decomposition [10, 11]. Then the minimal set of identifiable parameters are obtained known as base parameters represented by $\chi_b$.

$$\tau = Y_b \chi_b$$  \hspace{1cm} (5)$$

$Y_b$ is subset of the of independent columns of $Y$ of size (number of links $(n)$×number of base parameters).

An excitation trajectory used to get the joint angle $\theta$ of each joint, joint velocity $\dot{\theta}$ and acceleration $\ddot{\theta}$ must persistently excite the given robotic system. In this paper we employed the quintic trajectory inside the software environment of RoboAnalyzer which is discussed in Section 3.

### 2.3 Trajectory parameterization

The quintic trajectory for each joint is taken into account for joint values to be input in Equation (5). The joint position $\theta$, velocity $\dot{\theta}$ and acceleration $\ddot{\theta}$ for $i^{th}$ joint are,

$$\theta(t)=a_5t+a_4t^2+a_3t^3+a_2t^4+a_1t^5$$ \hspace{1cm} (6)$$

$$\dot{\theta}(t)=a_4+2a_5t+3a_6t^2+4a_7t^3+5a_8t^4$$ \hspace{1cm} (7)$$

$$\ddot{\theta}(t)=2a_4+6a_5t+12a_6t^2+20a_7t^3$$ \hspace{1cm} (8)$$

Note that for other trajectories, such as cubic polynomial trajectory acceleration cannot be specified at each point. Hence acceleration will be discontinuous and will result in impulsive jerk motion of the robot. Three constraints on position, velocity and acceleration at each point must be specified to avoid jerk. Therefore, a fifth order polynomial trajectory as in Equation (5) with different initial and final condition on joint angle motion was used in data acquisition method in Section 3.

### 3 IDENTIFICATION

#### 3.1 Data Acquisition from RoboAnalyzer

RoboAnalyzer is a 3D model based software which can be used for kinematic and dynamic analyses of a serial manipulator. The kinematic parameters can be given input in the form of DH parameters whereas the dynamic parameters for each link are given input in the form of six components of moment of inertia of a link about its center of mass with respect to link fixed frame, three components of position of center of mass in the link frame and mass. Table 1 lists the kinematic parameters of the 6R (R: revolute) spatial manipulator KUKA KR5 model. The kinematic model is shown in Figure 1(b).

<table>
<thead>
<tr>
<th>Link</th>
<th>DH parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.400 0.180</td>
</tr>
<tr>
<td>2</td>
<td>0.135 0.600</td>
</tr>
<tr>
<td>3</td>
<td>0.135 0.120</td>
</tr>
<tr>
<td>4</td>
<td>0.620 0</td>
</tr>
<tr>
<td>5</td>
<td>0 0</td>
</tr>
<tr>
<td>6</td>
<td>0 0</td>
</tr>
</tbody>
</table>

Table 1. Kinematic parameters of the robot

![a) CAD model of the KUKA KR5 b) Kinematic structure](image)

**Figure 1.** Kinematic model and DH parameters of the six DOF robot
Table 2. Dynamic parameter input in the RoboAnalyzer

<table>
<thead>
<tr>
<th>Link</th>
<th>$I_{xx}$</th>
<th>$I_{yy}$</th>
<th>$I_{zz}$</th>
<th>$I_{xy}$</th>
<th>$I_{yz}$</th>
<th>$I_{zx}$</th>
<th>$(x)_{C.G}$</th>
<th>$(y)_{C.G}$</th>
<th>$(z)_{C.G}$</th>
<th>$m$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit</td>
<td>Kg-m²</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>meter (m)</td>
<td>Kg</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0.322</td>
<td>-0.018</td>
<td>-0.145</td>
<td>0.467</td>
<td>-0.014</td>
<td>0.478</td>
<td>-0.091</td>
<td>-0.067</td>
<td>-0.006</td>
<td>26.980</td>
</tr>
<tr>
<td>2</td>
<td>0.541</td>
<td>0.000</td>
<td>-0.005</td>
<td>0.552</td>
<td>0.017</td>
<td>0.044</td>
<td>-0.333</td>
<td>-0.002</td>
<td>-0.039</td>
<td>15.920</td>
</tr>
<tr>
<td>3</td>
<td>0.775</td>
<td>-0.009</td>
<td>0.025</td>
<td>0.750</td>
<td>0.007</td>
<td>0.208</td>
<td>-0.032</td>
<td>0.008</td>
<td>0.034</td>
<td>25.852</td>
</tr>
<tr>
<td>4</td>
<td>0.010</td>
<td>0.002</td>
<td>0.000</td>
<td>0.020</td>
<td>0.000</td>
<td>0.024</td>
<td>0.000</td>
<td>-0.109</td>
<td>0.008</td>
<td>4.088</td>
</tr>
<tr>
<td>5</td>
<td>0.002</td>
<td>0.000</td>
<td>0.000</td>
<td>0.004</td>
<td>0.000</td>
<td>0.004</td>
<td>0.000</td>
<td>0.010</td>
<td>0.033</td>
<td>1.615</td>
</tr>
<tr>
<td>6</td>
<td>$6.20\times10^{-6}$</td>
<td>0.000</td>
<td>0.000</td>
<td>$3.01\times10^{-6}$</td>
<td>0.000</td>
<td>$3.29\times10^{-6}$</td>
<td>0.000</td>
<td>0.000</td>
<td>0.111</td>
<td>0.016</td>
</tr>
</tbody>
</table>

The three components of gravity vector were taken in the base frame. Once the parameters were fed into RA regarding joints trajectory for, the joint torques were then obtained using the inverse dynamics module in RA. The plots and the values of the joint angle, velocity, acceleration and torques can be obtained and saved from the graph section.

The joint values (angle, velocity and acceleration) were taken from the RoboAnalyzer at 500 time samples. Quintic motion was given to each of the joints. For joint 1, Figure 2 shows the quintic trajectory input to the joints. The initial conditions for the trajectory were, $\theta_0 = 0, \dot{\theta}_0 = 30, \ddot{\theta}_0 = 0, \theta_1 = 0, \dot{\theta}_1 = 0$, and $\ddot{\theta}_1 = 0$ for joint 1. For joint 2 to joint 5, the values of final joint angle were taken as 45, 60, 75, 90, and 105 respectively. Each joint was excited with quintic trajectories and the joint torque values were generated using inverse dynamics module of RA and then the data points were saved from the graph section in .csv format. The screen shot from the RA for different trajectories that are available is shown in Figure 2(b) and by right clicking on graph we can export the data in .csv format for further analysis in computational software like Matlab. More help on RA can be found in its user manual [9].
3.2 Reduced Dynamic Model

The mathematic model obtained from the Newton Euler method discussed in Section 2 is in terms of inertial parameters listed in Equation 1. The reduced dynamic model neglects those parameters which does not affect the system dynamics and further combines the remaining into base parameter set. The quintic trajectory and torque values obtained from RA were utilized to obtain the base parameters of the robot as discussed in Section 2. A total of 36 base parameters were formed which are listed in Appendix 2. Out of sixty SIP for six link robot, we have deduced 36 base parameters in the reduced model using quintic trajectories.

3.3 Results and Validation

To validate the reduced model, a general cycloidal trajectory [3] as in Equation (9) was given as input in Equation 5 and the corresponding torques were estimated.

\[
\theta = \theta(0) + \frac{\theta(T) - \theta(0)}{T} \left[ T - \frac{T}{2\pi} \sin \left( \frac{2\pi T}{T} \right) \right]
\]  

(9)

The same trajectory was given in RoboAnalyzer to obtain the joints torques. Figure 3 shows the plots of estimated joint torques using identified model and the joint torques obtained from RoboAnalyzer.

- [Image of joint torques]

Figure 2. Input trajectory and the corresponding torque value obtained using Inverse Dynamics in RoboAnalyzer for joint 1.
We found that the identified dynamic model torques for a general trajectory closely matched with the RA torques values. Joint 4 and 6 torques as in Figure 3(d) and (f), are negligible, this is due to the fact that the mass and inertial values of these two axes as listed in Table 2 are also negligible. Major torques are coming at joint 2 and 3. The reduced model is generally implemented in robot controller to compute torque required for a given trajectory. The close match is obtained in Figure 3, since the measurement noise in recording the angle and torque values are not coming into picture in the simulated results obtained from RA. In future the effect of noise can also be analysed by adding Gaussian noise to the joint and torque values obtained from RA.

4 CONCLUSIONS

The reduced Dynamic model of a six degree of freedom serial manipulator was formulated in terms of regressor matrix and base parameters. The base parameters were identified for the six DOF KUKA KR5 serial manipulator in a simulated environment of RoboAnalyzer. Quintic trajectory was used to get the reduced dynamic model with base parameters. A total of 36 base parameters were identified in the reduced dynamic model. The reduced dynamic model torque values and the RA torque values for a general cycloidal trajectory were closely in match with each other. In future the data from the installed robot KUKA KR5 will be used to perform identification giving possible excitation trajectory input to the robot controller.

5 ACKNOWLEDGEMENT

The financial support to the first author from the sponsored project by BRNS/BARC Mumbai under the “Programme for Autonomous Robotics Lab” at IIT Delhi is sincerely acknowledged.
REFERENCES


APPENDIX

This section lists the kinematic definition and the geometry of the general links. The base parameters are also listed in Appendix 2.

APPENDIX 1

In this section the configuration and dimension of the robotic mechanism, in body fixed frames are defined in Figure A1. In this case the robotic system considered only consists of the revolute joint.

Table A1. DH parameters with their definition

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( b_i )</td>
<td>Joint offset</td>
<td>( X_i \downarrow \text{distance} \atop @ Z_i \rightarrow X_{i+1} )</td>
</tr>
<tr>
<td>( \theta_i )</td>
<td>Joint Angle</td>
<td>( X_i \uparrow \text{rotation, ccw} \atop @ Z_i \rightarrow X_{i+1} )</td>
</tr>
<tr>
<td>( a_i )</td>
<td>Link Length</td>
<td>( Z_i \downarrow \text{distance} \atop @ X_{i+1} \rightarrow Z_{i+1} )</td>
</tr>
<tr>
<td>( \alpha_i )</td>
<td>Twist Angle</td>
<td>( Z_i \uparrow \text{rotation, ccw} \atop @ X_{i+1} \rightarrow Z_{i+1} )</td>
</tr>
</tbody>
</table>

*In the table read symbol \( \rightarrow \) as “to”, \( \downarrow \) as perpendicular”, \( \atop \) as “about or along”

Figure A1. Coordinate frames and associated parameters
The inertia tensor of the rigid body about the link frame is given in Equation (A1) and its representation in different reference frame is expressed by it with the rotation matrix as in Equation (A2)

\[
\mathbf{I}^O = \mathbf{I}^C - m (\mathbf{r} \times \mathbf{r}) \tag{A1}
\]

\[
[I^O]_{ij} = Q_i Q_j^T \tag{A2}
\]

**APPENDIX 2**

Base parameters and their identified values are listed below:

**Table A2.** Base parameters for the KUKA KR5 CAD model robot in RA

<table>
<thead>
<tr>
<th>Base Parameters Expressions</th>
<th>Identified Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>([I_{x3} + I_{x2} + I_{x1} + 0.0342(m_1 - m_4 - m_3 - m_2) - 0.309m_1 + 0.36m_1x_1 - 0.27m_2z_2]</td>
<td>-12.2955</td>
</tr>
<tr>
<td>([I_{y3} + I_{y2} + I_{y1} + 0.36(m_3 + m_4 + m_5 + m_6)]_1</td>
<td>15.3204</td>
</tr>
<tr>
<td>([I_{z3} + I_{z2} + I_{z1} + 0.081(m_1 - m_4 - m_3 - m_2) + 0.6m_1y_1 - 0.6m_2z_2]</td>
<td></td>
</tr>
<tr>
<td>([I_{x2} + I_{x1} + 0.36(m_2 - m_5 - m_2 - m_1)]_1</td>
<td></td>
</tr>
<tr>
<td>([I_{y3} - I_{y2} + I_{y1} + 0.0144m_3 + 0.3988(m_1 + m_3 + m_5) + 1.24m_4y_4]</td>
<td>2.7435</td>
</tr>
<tr>
<td>([I_{z3} + I_{z2} - 0.12m_3y_3]</td>
<td></td>
</tr>
<tr>
<td>([I_{x1} + I_{x2} - 0.0144m_1 + 0.37m_1 + 0.37m_2 + 1.24m_3y_3]</td>
<td></td>
</tr>
<tr>
<td>([0.12(m_3 + m_4 + m_5 + m_6) + m_1x_1]</td>
<td>2.9612</td>
</tr>
<tr>
<td>([0.62(m_4 + m_4 + m_6)m + m_1y_3 + m_3z_3]</td>
<td>3.9791</td>
</tr>
<tr>
<td>([I_{x3} + I_{y3} + I_{z3}]_{16}</td>
<td>-0.0996</td>
</tr>
<tr>
<td>([I_{y3} + I_{x3}</td>
<td>0.0244</td>
</tr>
<tr>
<td>([m_3z_3 - m_1y_3]</td>
<td></td>
</tr>
<tr>
<td>([I_{x3} + I_{y3} - I_{z3} + 0.0132m_6 + 0.23m_6z_6]</td>
<td>0.0005</td>
</tr>
<tr>
<td>([I_{y3} + I_{x3} + 0.013225m_6 + 0.23m_6z_6]</td>
<td></td>
</tr>
<tr>
<td>([0.115m_6 + m_6z_6 + m_6z_6]</td>
<td></td>
</tr>
<tr>
<td>([I_{z3} - I_{y3} - I_{x3}</td>
<td></td>
</tr>
<tr>
<td>([I_{y3}]_3</td>
<td></td>
</tr>
<tr>
<td>([I_{z3}]_3 = 0.01575, [m_3y_3]</td>
<td></td>
</tr>
<tr>
<td>([I_{y3}]<em>3 = 3.17 \times 10^{-3}, [I</em>{y3}]_7</td>
<td></td>
</tr>
<tr>
<td>([I_{z3}]<em>3 = 0.0035, [I</em>{y3}]<em>5 = -5.33 \times 10^{-4}, [I</em>{z3}]_3 = 3.29 \times 10^{-6}</td>
<td></td>
</tr>
</tbody>
</table>

Magnitude of the parameters below are are numerically equicvalent to machine zero, which was taken as order of 10^-10 in simulation.

\([I_{x3}]_3, [m_6x_6]| 3.2 \times 10^{-6} |
\([I_{y3}]_3, [m_6x_6]| 3.2 \times 10^{-6} |
\([I_{z3}]_3, [m_6x_6]| 3.2 \times 10^{-6} |
\([I_{y3}]_3, [m_6x_6]| 3.2 \times 10^{-6} |
\([I_{z3}]_3, [m_6x_6]| 3.2 \times 10^{-6} |

Note that the index in the subscript above ([ ]_1) shows the base parameter number.
Energy-Efficient Point-to-Point Trajectory Generation for Industrial Robotic Machines

P. Boscariol *, G. Carabin †, A. Gasparetto*, N. Lever †, R. Vidoni†

* DIEG-Dipartimento di Ingegneria Elettrica, Gestionale e Meccanica, University of Udine
Via delle scienze, 33100, Udine (I)
paolo.boscariol@uniud.it
gasparetto@uniud.it

† Faculty of Science and Technology Free University of Bolzano
Piazza Università, 39100, BZ (I) renato.vidoni@unibz.it
nikola.lever@gmail.com Giovanni.Carabin@unibz.it

ABSTRACT
Robots and mechatronic applications are widely used for process automation in plants and factories. Trajectory planning is a fundamental issue for operating these industrial machines and, in this work, a point-to-point (PTP) trajectory based on a S-curve has been designed to reduce the consumed energy of a typical mechatronic system, i.e. a robotic linear axis made of an electric-motor that moves a payload on a plane by means of a transmission system and a toothed belt. This evaluation is used to find the minimum energy consumption conditions, also taking into account the possibility of using a regenerative brake. The problem is defined and solved for several operative conditions, either in a closed-form or numerically using a genetic algorithm.

Keywords: optimum trajectory planning, energy efficiency, industrial robotic machines, point-to-point, energy recovery.

1 INTRODUCTION
Machine tools, cranes, material handling, robots and, more in general, mechatronic applications are widely used for process automation in plants and factories. Trajectory planning is a fundamental issue for controlling these industrial machines [1].

In the past, the focus has been set mainly on the solution of minimum time problems, i.e. the main objective has been to find the solution that leads to a lower execution time of a task. Later the attention has been cast also to the problem of jerk minimization, which is aimed at the reduction of the vibration phenomena and of the mechanical failure risks [2, 3, 4, 5, 6].

Recently, in the wake of the increasing of the prices of energy and the popularity of energy-saving strategy, the concept of energy efficiency has spread also to industrial applications. In this perspective, the development of energy efficient trajectories is a very promising technique, since it can be effectively applied also to systems that do not include any regenerative braking system, i.e. to the majority of the equipment currently in use. Moreover, more recent technological improvements such as variable frequency devices and energy recovery systems are setting new paths for energetic reduction of industrial applications.

A fundamental distinction, among the available methods adopted for reducing energy consumption by an optimum trajectory planning approach, is the exploitation of a model-based, e.g. [7, 8], or of a model-free approach, e.g. [5]. Model-free approaches are appealing for their generality and their high portability on several different industrial applications. On the other hand, model-based approaches can achieve better results in specific cases. Luckily, many mechatronic industrial systems can be adequately modeled by mathematic models. In [9], an energy saving controller with a model based approach through a typical dynamics of feed drive systems including an inertia term, a viscous friction term, a Coulomb friction term and a back-EMF term is introduced. In [10], a model-based method for reducing the total energy consumption of pick-and-place manipulators for given TCP position profiles by a time-scaling approach is presented. In [11], the simultaneous evaluation of both the energy efficiency and the smoothness in the most significant off-line
non-model based methods and algorithms currently adopted in industrial applications is presented. Focusing on the most recent works, the minimum energy trajectory optimization is treated considering the electrical energy exchange via the shared inverter DC link, thus allowing to find a different energy minimum with respect to the available literature approaches [12, 13].

In this work, the design of a point-to-point (PTP) trajectory, known as the S-curve trajectory, to reduce the consumed energy of a typical mechatronic system, i.e. a robotic linear axis made of an electric-motor that moves a payload on a plane by means of a transmission system and a toothed belt, is addressed. The chosen trajectory is a simple motion trajectory (so as to minimize controller and implementation costs), which consists of an acceleration period, a constant velocity period and a deceleration period. The minimum of the required energy is studied, evaluated and found either in a closed form by mathematical methods or numerically through the use of genetic algorithms. In such a way, the S-curve values that ensure the best results in terms of energy savings are identified under different working conditions, e.g. gravity.

The paper is organized as follows: the energy saving trajectory design is presented in Section 2; Section 3 reports the modeling of the robotic linear axis while in Section 4 the minimum absolute-energy problem is formulated by minimizing the energy consumption.

2 S-CURVE TRAJECTORY

The S-curve trajectory is widely used in industrial applications because it is simply implementable in old and new controllers and assures continuous accelerations and smooth jerk profiles [14].

The S-curve trajectory is implemented through a cycloid motion during the acceleration and deceleration periods that permit to have a limited acceleration value at the start and end of the motion. It is basically described by four parameters: acceleration time, constant velocity time, deceleration time and constant velocity magnitude. In eq. (1), (2) the S-curve trajectory is described in terms of velocity and acceleration.

\[ v(t) = \begin{cases} 
\frac{v_0}{2}(1 - \cos(\omega_1 t)) & \text{if } t \in [0, t_1), \\
v_0 & \text{if } t \in [t_1, t_1 + t_2), \\
\frac{v_0}{2}(1 + \cos(\omega_3 t')) & \text{if } t \in [t_1 + t_2, t_1 + t_2 + t_3), 
\end{cases} \]

\[ a(t) = \begin{cases} 
\frac{v_0}{2}(\omega_1 \sin(\omega_1 t)) & \text{if } t \in [0, t_1), \\
0 & \text{if } t \in [t_1, t_1 + t_2), \\
\frac{v_0}{2}(\omega_3 \sin(\omega_3 t')) & \text{if } t \in [t_1 + t_2, t_1 + t_2 + t_3), 
\end{cases} \]

where \( t_1, t_2, t_3 \) and \( v_0 \) are the acceleration time, the constant velocity time, the deceleration time and the maximum velocity, respectively. \( t' = t - t_1 - t_2, \ \omega_1 = \pi/t_1, \ \omega_3 = \pi/t_3. \)

If \( t_1 \neq t_3 \), it is possible to define the following relations:

\[ v_0 = 2L/(t_2 + 2t_2 + t_3), \quad t_2 = T - t_1 - t_3 \]

Fig. 1 on the following page shows the S-curve position, velocity and acceleration profiles for \( t_1 = t_3 = 0.5s \) and the total motion time \( T = 3s \).

3 ELECTRO-MECHANICAL MODEL OF A ROBOTIC LINEAR AXIS AND ENERGY FORMULATION

The system of choice is composed on an electric motor that moves a payload, e.g. a cartesian robot axis, in a horizontal or vertical plane by means of a toothed belt with a reduction ratio equal to \( K_r \) (Fig. 2 on the next page). The model takes into account the load inertia, the viscous and Coulomb friction, as well as the resistive loss in the motor windings. According to the S-curve trajectory, the motor torque necessary to move the payload is described by the equation (3):
\[ \tau(t) = \left[ \frac{j_m}{K_r} + mK_i \right] a_i(t) + [DK_r] v_i(t) + [mgK_r \sin \varphi + T_aK_r] \]  

(3)

where \( j_m \) is the motor moment of inertia, \( m \) the load (l) mass, \( D \) the viscous friction coefficient, \( \varphi \) the inclination angle and \( T_a \) the dynamic friction.

Therefore, the instantaneous current and voltage in the motor phase are:

\[
i(t) = \frac{1}{K_i} \left\{ \left[ \frac{j_m}{K_r} + mK_i \right] a_i(t) + [DK_r] v_i(t) + [mgK_r \sin \varphi + T_aK_r] \right\}
\]

(4)

\[
e(t) = Ri(t) + \frac{K_e}{K_r} v_i(t)
\]

(5)

with \( K_i \) the torque constant and \( K_e \) the back-EMF constant.

Then, the instantaneous power can be expressed as:

\[
P(t) = Ri(t)^2 + \frac{K_e}{K_r} v_i(t)i(t)
\]

(6)
Eq. 6 is composed of two terms: the first one is the power loss in the motor winding while the second one is the power used to move the payload. If the latter is positive, the system is in direct motion (i.e. the motor is providing energy to the load), otherwise retrograde motion occurs. In the latter situation the drive system can recover energy since the motor is actually working as a brake (i.e. it is providing a negative work). Starting from the power formulation in eq. 6, it is possible to determine the overall instantaneous energy as:

\[
E = E_{\text{res}} + E_{m1} + E_{m2}
\]

where \(E_{\text{res}}\) is the energy loss due to the motor windings, \(E_{m1}\) the energy consumed to move the load \((E_{m1} > 0)\) and \(E_{m2}\) the energy recovered when the motor acts as a generator \((E_{m2} < 0)\).

A simpler method to calculate the total energy, therefore without taking into consideration the three contributions as separated, is to compute the integral:

\[
E = \int e(t) i(t) dt
\]

Now, following the approach in [9], it is possible to find the energy formulation for the modeled mechanical system in the generic case \(t_1 \neq t_3\) as:

\[
E = \frac{\left(\frac{1}{2} c_1 \pi^2 + \frac{1}{2} c_1 \pi^2 + 4c2(T - t_1 - t_3) + \frac{3}{2} \pi c2t_1 + \frac{3}{2} \pi c2t_1 \right) L^2}{(2T - t_1 - t_3)^2} + c3L + c4T \tag{7}
\]

where:

\[
\begin{align*}
b1 &= (mg + T_d)K_r/K_t, \\
b2 &= DK_r/K_t, \\
b3 &= (j_m/K_r + mK_r)/K_t, \\
b4 &= Rb1, \\
b5 &= Rb2 + K_e/K_r, \\
b6 &= Rb3, \\
c1 &= b3b6, \\
c2 &= b2b5, \\
c3 &= b1b5 + b2b4, \\
c4 &= b1b4, \\
c5 &= b1b6 + b3b4, \\
c6 &= b2b6 + b3b5.
\end{align*}
\]

4 ENERGY CONSUMPTION OPTIMIZATION

Once fixed the motion displacement, the S-curve trajectory is described by three parameters: the acceleration time, the deceleration time and the total motion time.

In the following, the minimum of the energy consumption is addressed and found, either in a closed-form or through a genetic algorithm, acting on the three variables.

Three main cases for the three variables are studied:

- energy optimization with \(t_1 = t_3\) free, total time fixed;
- energy optimization with \(t_1 = t_3\) and total time free;
- energy optimization with free \(t_1, t_3\) and total time.

4.1 Energy optimization with \(t_1 = t_3\) free, total time fixed

The energy minimization in this first case is the easiest one. Indeed, it is possible to obtain a simple energy formulation, shown in eq. (8), starting from the eq. (7), by forcing \(t_1 = t_3\).

\[
E = \frac{-5c2\pi^2 + 4c2Tt_1 + c1\pi^2}{4t_1(T - t_1)^2} L^2 + c3L + c4T \tag{8}
\]

The consumed energy depends only on the acceleration time \(t_1\). To find a minimum or a maximum for the equation, the first time-derivative of eq. (8) has to be computed, eq. (9).
Table 1: Test-case mechanical and electrical parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>mass</td>
<td>m</td>
</tr>
<tr>
<td>track length</td>
<td>L</td>
</tr>
<tr>
<td>transmission ratio</td>
<td>K_r</td>
</tr>
<tr>
<td>torque constant</td>
<td>K_t</td>
</tr>
<tr>
<td>back-EMF constant</td>
<td>K_e</td>
</tr>
<tr>
<td>motor moment of inertia</td>
<td>j_m</td>
</tr>
<tr>
<td>electrical motor resistance</td>
<td>R</td>
</tr>
<tr>
<td>gravity acc.</td>
<td>g</td>
</tr>
<tr>
<td>viscous friction</td>
<td>D</td>
</tr>
<tr>
<td>dynamic friction</td>
<td>T_a</td>
</tr>
<tr>
<td>inclination angle</td>
<td>θ</td>
</tr>
</tbody>
</table>

![Figure 3](image.png)

(a) energy first derivative, \( b_3 = 10 \).  
(b) energy first derivative, \( b_3 = 1 \times 10^{-7} \).

\[
\frac{dE}{dt_i} = \frac{(-5c2T^2 + 3c2T_1^2 + 3c1\pi^2T_1 - c1T^2)L^2}{2t_i^3(T - t_i)^3} = 0
\]  

(9)

It is worthwhile to highlight how the first derivative of the energy is heavily affected by the value of \( b_3 = (j_m/K_t + m*K_r)/K_t \) that appears in the \( c_1 \) term. In Fig. 3, the first time derivative of the energy is shown for two different \( b_3 \) values; by inspecting the graph and by taking into consideration the definition of \( b_3 \), it can be seen that if the load inertia is relatively smaller than the torque constant, i.e. \( b_3 \) is small, the first time derivative of the energy is always positive. Thus, the energy function decreases with the increase of the acceleration time. If a \( \lambda \) term equal to the ratio between the acceleration time \( t_1 \) and the total time \( T \) is introduced, the optimum energy value is found by imposing \( \lambda \), and thus the acceleration time, as small as possible.

In the other case, if \( b_3 \) is big enough, see the plot in Fig. 3(a), the first energy derivative changes both its magnitude and its sign during the acceleration time. Hence, a point of minimum in the energy function can be found at \( \lambda = 1/3 \). In this work, \( b_3 \) is considered big enough to have a point of minimum for the energy function.

The proposed optimization methods are tested on a benchmark system, whose physical values are shown in Table 1. To better analyze the energy consumption, attention has to be paid to determine and evaluate the energy consumption both in the total motion displacement and in each of the three parts that compose the S-curve trajectory, i.e. during the acceleration phase, the constant velocity phase and in the deceleration phase.
As shown in Fig. 4, if the $\lambda$ value is small, the energy consumption during all the three time-blends is high and the energy required during the constant velocity time decreases when $\lambda$ increases. On the other hand, if the value of $\lambda$ is high, the energy of the other two blends increases and vice-versa for low $\lambda$ values. Thus, the sum of the three parts confirms the minimum energy condition with $\lambda = 1/3$.

![Figure 4: Total energy, energy during acceleration period, energy during constant velocity period, energy during deceleration period, at $\lambda$ varying, with $t_1 = t_3$, $T = 3s$.](image)

Assuming the displacement in the horizontal plane, i.e. without the effect of the gravity force, it is possible to investigate the energy consumption or the energy recoverable, with respect to the $\lambda$ value.

In Fig. 5 on the next page, the different energy contributions are shown. For each value of $\lambda$, part of the energy is consumed or lost and part is recovered during the load deceleration phase. The main issue in reducing the energy wasted is to reduce the energy used and lost in the armature resistance. Indeed, the global consumption for all the $\lambda$ values, overlaps the energy dissipations in the resistance.

### 4.2 Energy optimization with $t_1 = t_3$ and total time free

The energy minimization in the second case allows to change the acceleration time, the deceleration time and the total time as well. The only binding assumption is that the acceleration time has to be equal to the deceleration time. So, the set of variables to be optimized includes $t_1$ and $T$.

Again, the energy formulation to be taken into consideration is given by eq. (7). By computing the points of maximum and minimum:

$$
\frac{dE}{dt_1} = \left(\frac{-1}{2}\right)E^2 \left(-3c_1\pi^2 t_1 + 5c_2 t_1^3 - 3T c_2 t_1^2\right) = 0
$$

$$
\frac{dE}{dT} = c_4 - \frac{\omega_0^2 c_2^2}{t_1 (T-t_1)} = 0
$$

(10)

the Hessian with respect to the variables $t_1$ and $T$ can be evaluated and the minimum energy solution can be found. By performing all the steps of the method, the point of minimum $\lambda = 0.32$ and $T = 4.07s$ is obtained.

The graph in Fig. 6 on the following page highlights the effects of varying both $t_1$ and $T$ on the total energy consumption. To summarize, in the case that $t_1 = t_3$ and with free total execution time
$\lambda = \frac{t_1}{T} = \frac{t_3}{T}$

$E = \int_{0}^{T} \frac{1}{2} m v^2 dt$

$T$, the energy efficiency is maximized for when the acceleration, the deceleration and the constant velocity phases have equal duration.

**Figure 5:** Energy dissipation on the resistance, energy consumed to move the load, energy recovered, with respect to $\lambda$. The vertical black line represents the energy minimum value.

**Figure 6:** Energy consumption with respect of $t_1$ for different $T$ slopes.

### 4.3 Energy optimization with free $t_1$, $t_3$ and total time

The energy minimization in this case takes into account the most general condition, i.e. the one in which $t_1$, $t_3$ and $T$ are free.

The problem has to deal with the optimization of an equation in three variables. The optimization problem is in this case solved numerically, through a genetic algorithm procedure. The genetic algorithm (GA) optimization belongs to the larger class of evolutionary algorithms where a population of candidate solutions to an optimization problem is evolved toward better solutions. Here, the "ga" MATLAB function has been exploited for finding a local minimum of an object function $fitnessfcn$ under linear bounds on $t_1$, $t_3$ and $T$. The solution of the optimization problem, i.e. the minimum overall power consumption, is achieved for: $\lambda_1 = 0.31$, $\lambda_3 = 0.34$ and $T = 4.07s$. 

1431
In this case, the optimum total motion time $T$ is higher than the previously fixed one, i.e. $3s$, demonstrating how it is possible to reduce the energy consumption with the three variables optimization. However, as in the previous cases, the optimum values for the acceleration, deceleration and constant velocity phases are very near to the $1/3$ condition allowing to assess that the equal distribution of the phase along the whole motion time allows to obtain at least a quasi minimum energy consumption.

4.3.1 Gravity effect

If the slope angle $\theta$ changes from a flat condition up to a vertical condition, the energy $E$ increases. This can be seen looking at the current eq. (4), as shown in Fig. 7. Since the gravity effect does not depend on the total time $T$, the acceleration $t_1$ and deceleration $t_3$ time but only on the configuration, i.e. the slope angle $\theta$, the minimum energy configuration does not change.

![Figure 7: Trend of energy consumption at $\lambda$ and $\theta$ varying, with $t_1 = t_3$ and $T = 3s$. The vertical black line represents the energy minimum value.](image)

5 CONCLUSIONS

In this paper, starting from the results in [9], the issue of finding a minimum energy consumption for a point-to-point motion on a typical mechatronic system, i.e. a robotic linear axis, has been addressed. By taking into consideration the S-curve primitive, the expression of the total energy consumption for point-to-point motion has been formulated and computed. Then, the parameters that achieve minimum energy consumption, given the possibility to recover energy when braking, have been found in different cases either in a closed form or through a numerical solution. Future work will cover both the experimental validation of the method and the extension of the proposed solution to multi-axis systems.

References


Exploiting the equations of motion for biped robot control with enhanced stability

Johannes Mayr, Alexander Reiter, Hubert Gattringer, Andreas Müller

Institute of Robotics
Johannes Kepler University Linz
Altenbergerstr. 69, 4040 Linz, Austria
[joh.mayr, alexander.reiter, hubert.gattringer, a.mueller]@jku.at

ABSTRACT

The scope of the present paper is the derivation of the equations of motion for humanoid robots, in particular legged robots. The derivation is performed in a modular and structured manner and it is shown how these equations can be exploited for the control of biped robots. The used methods allow to easily adopt the kinematic structure of single limbs and to reuse results obtained for limbs with similar kinematic structure but different inertial parameters such as in case the left leg is just a mirrored version of the right one. After finding a recursive formulation to calculate the equations of motion we perform various state transformations and apply some model simplifications to obtain expressions that can be used to effectively solve control problems. Two applications, compensating for the overall angular momentum and calculation of feed-forward torques, are shown. In both applications we can exploit the recursive calculation of the equations of motion used during the subsystem synthesis giving rise to real-time algorithms that can be used on a physical humanoid robot system.

Keywords: multi-body modeling, angular momentum, inverse dynamics, biped control.

1 INTRODUCTION

When stabilizing a biped humanoid robot system one has to cope with many different control challenges mainly introduced by the high amount of degrees of freedom and the underactuation of the system. To allow the robot to walk stably without tripping over one has to consider the contact forces between the robot and the environment. Typically the stabilization of the robot and the generation of the walking pattern is treated separately ([4, 3, 7]) which allows to use models of different complexity for the different tasks. First a set of trajectories is defined according to the desired walking behavior ([4, 2, 6]). This trajectory is then tracked as accurately as possible. In the ideal case the robot would be able to walk dynamically balanced just by tracking those trajectories, but due to the unknown environment, inaccurate parameters and external disturbances different control layers have to modify the generated walking pattern to stabilize the robot around the given equilibrium trajectory. Especially the walking pattern generation relies on an accurate system model given by the equations of motion for the multi-body system [2, 6]. Depending on the desired walking behavior (e.g. fast walking) more accurate models need to be used by the walking pattern generator to provide physically consistent trajectories. On the other hand, complex models are computationally heavy and algorithms based on them may not be executed in realtime. Thus one needs to find a reasonable trade-off between model accuracy and computational effort of the model. Various different models have been proposed in the literature. The simplest one is given by a single pointmass at constant height and is typically refered to as three dimensional linear inverted pendulum mode (3D-LIPM) [4]. Various offline walking pattern generators use the full or a slightly reduced multi-body model of the robot [2]. In this paper we are going to present various different models that can be used for the control of a biped robot.

After a short introduction to the used multi-body algorithm a detailed multi-body model and various simplified models of the robot will be derived in Section 2. In Section 4 these models will then be used to exemplary show two possible control applications. Section 5 will finally conclude the paper.
1.1 Coordinates and Frames

The frames and their origin used during the following sections are summarized in Fig. 1. The frame $W$ is a world fixed frame with the $z$-axis pointing in the opposite direction of the gravity vector $g$. A body fixed frame $B$ is attached to the torso. For the description of the position of the limbs the frames $R, L$ are fixed to the left leg and the right leg. The twist between two frames is given by $V_{B}^L \in \mathbb{R}^6$ and corresponds to the linear and angular velocity of the frame $L$ relative to frame $B$ represented in $L$. The superscript $\{\}_b$ denotes a body twist.

1.2 Configuration Space

The kinematic configuration of robotic manipulators is typically described by their joint positions. For a humanoid robot this is not sufficient. Depending on the contact state of the robot we need some additional information to be able to fully define the position of the robot in the world frame. In the worst case, there is no contact of the robot to the ground. This state will be referred as free floating base. Thus the configuration space is given by

$$Q = \{(H_B, q_J) | H_B \in SE(3), q_J \in J\}$$

(1)

with $J = J_1 \times \ldots \times J_n$ as the joint space of a robot with $n$ drive units and $J_i \subseteq \mathbb{R}^m$ as the parameter space representing the $m$ degrees of freedom of the $i$-th drive unit. Further $q_J$ summarizes all joint positions. The generalized velocities corresponding to the configuration space $Q$ are chosen as

$$\dot{s} = \begin{bmatrix} V_B^T & \dot{q}_J^T \end{bmatrix}^T.$$  

(2)

1.3 Task Space

To simplify trajectory and controller design it is useful to define different task spaces for the robot:

$$T_1 = \{(r_c, R_B, q_J) | r_c \in \mathbb{R}^3, R_B \in SO(3), q_J \in J\},$$

(3)

$$T_2 = \{(r_c, R_B, H_R, H_L, \dot{q}_J) | r_c \in \mathbb{R}^3, R_B \in SO(3), H_i \in SE(3), \dot{q}_J \in \dot{J}\}$$

(4)

with $\dot{J} \subset \dot{J}$. While $T_1$ still uses the joint positions, different to $Q$ now the inertial CoM position $r_c$ is used to describe the position of the robot in space. The orientation of the upper-body is given by the rotation matrix $R_B$. In $T_2$ the joint positions of the legs are replaced by the homogeneous transformations to the feet. The joint positions of the arms $\dot{q}_J$ remain in the task space.
The corresponding velocities are chosen to

\[ \dot{z}_1 = [\dot{r}_c^T \quad \dot{\omega}_B^T \quad \dot{q}_f^T]^T, \] (5)

\[ \dot{z}_2 = [\dot{r}_c^T \quad \dot{\omega}_B^T \quad (\dot{V}^B_R)^T \quad (\dot{V}^B_L)^T \quad \dot{q}_f^T]^T \] (6)

with the angular velocity of the upper body \( \omega_B \).

2 DYNAMIC MODELING

2.1 Subsystem Modeling

In general the equations of motion for a robot with \( n \) degrees of freedom and a free floating base is given in the form

\[ M(q) \ddot{s} + G(q, \dot{s}) \dot{s} - Q(q) = 0 \] (7)

where \( q \in \mathcal{D} \) is the tuple of generalized coordinates. The generalized non-holonomic velocities \( \dot{s} \in \mathbb{R}^{6+n} \) are given by Eq. 2. \( M \in \mathbb{R}^{(6+n) \times (6+n)} \) denotes the mass matrix, \( G \in \mathbb{R}^{(6+n) \times (6+n)} \) is the matrix of the centrifugal and Coriolis forces and \( Q \in \mathbb{R}^{6+n} \) is the vector of generalized forces and torques. In this work an approach based on the Projection Equation introduced in [1] is used to derive the multi-body dynamics from Eq. 7. For a multi-body system with \( N \) bodies the equations of motion are given by

\[ \sum_{i=1}^{N} \left( \frac{\partial g_i}{\partial s} \right)^T \left( \frac{\partial g_i}{\partial s} \right) \left[ R \dot{p} + R \dot{\omega} I_{RR} \dot{p} - \dot{r}_c^f \right] = 0. \] (8)

with \( R \dot{p}_i = m_i \dot{v}_{c,i} \) being the vector of linear momentum and \( R \dot{L}_i = R J_{c,i} \dot{\omega}_i \) the vector of angular momentum. The absolute velocity of the CoM and the angular velocity of the \( i \)-th body are given by \( \dot{r}_c, \dot{\omega}_i \in \mathbb{R}^3 \). The inertia tensor of the \( i \)-th body is given by \( R J_{c,i} \in \mathbb{R}^{3 \times 3} \). The angular velocity of the reference frame for the \( i \)-th body is given by \( R \dot{\omega}_{IR,i} \in \mathbb{R}^3 \). External actions acting on the CoM of the \( i \)-th body are considered by \( \dot{r}_c^f \in \mathbb{R}^3 \) and \( \dot{r}_c^f \in \mathbb{R}^3 \). The \( \tilde{\cdot} \) operator describes the skew symmetric matrix for calculating the cross product \( (\tilde{a}b = a \times b, \tilde{a} \in so(3)) \). For every body \( i \) in Eq. 8 a different arbitrary reference frame \( R \) can be used. For the proposed system and for robotic systems in general, it is useful to combine interconnected bodies to subsystem assemblies (e.g. motor - gear - structural components). This can be done by partitioning the sum of Eq. 8 into \( N_t \) subsystems and splitting up the partial derivative

\[ \sum_{i=1}^{N_t} \left( \frac{\partial g_i}{\partial s} \right)^T \left\{ \sum_{j=1}^{N_t} \left[ \frac{\partial g_j}{\partial s} \right]^T \left[ R \dot{p} + R \dot{\omega} I_{RR} \dot{p} - \dot{r}_c^f \right] \right\} = 0. \] (9)

where every subsystem \( i \) consists of \( N_t \) bodies and has \( n_i \) degrees of freedom. Every subsystem is described by a different set of describing velocities \( \dot{y}_i \in \mathbb{R}^{6+n_i} \) which are themselves a function of the generalized velocities \( \dot{s} \). After summing up all bodies of a subsystem we obtain a representation for every subsystem and Eq. 9 can be rewritten to

\[ \sum_{i=1}^{N_t} \left( \frac{\partial y_i}{\partial s} \right)^T \{ M_i(q_s) \ddot{y}_i + G_i(q_s, \dot{y}_i) \dot{y}_i - Q_i(q_s) \} = 0. \] (10)

where \( M_i \in \mathbb{R}^{(6+n_i) \times (6+n_i)} \) represents the mass matrix of the subsystem, \( G_i \in \mathbb{R}^{(6+n_i) \times (6+n_i)} \) summarizes centrifugal and Coriolis forces and \( Q_i \in \mathbb{R}^{6+n_i} \) represents generalized forces and torques acting on the subsystem. The subsystem matrices and vectors are functions of the generalized velocities \( \dot{y}_i \) and the generalized coordinates \( q_s \in SE(3) \times \mathbb{R}^{n_v} \) of the subsystem. Both of them can be expressed as a function of the generalized velocities \( \dot{s} \) and the generalized coordinates \( q \) of the whole robotic system. After we found the corresponding matrices \( M_i \) and \( G_i \) and the vector \( Q_i \) from Eq. 10 by evaluating the terms in the curly brackets in Eq. 9 for every subsystem, they can
be combined to yield the equations of motion of the complete system. The outer sum in Eq. 9 can be rewritten as

$$\left[ \begin{array}{c} \frac{\partial y_1}{\partial s} \\ \vdots \\ \frac{\partial y_{N_s}}{\partial s} \end{array} \right]^T = \begin{bmatrix} M_1\ddot{y}_1 + G_1\dot{y}_1 - Q_1 \\ \vdots \\ M_{N_s}\ddot{y}_{N_s} + G_{N_s}\dot{y}_{N_s} - Q_{N_s} \end{bmatrix} = 0. \quad (11)$$

### 2.2 Assembly Strategy

In the case of the considered biped robot one reasonable choice of a decomposition in subsystems is given by Fig. 2. In total four different kinds of subsystems (subsequently referred to as limb subsystems) for the legs, the arms, the head and the torso are considered. While the leg and the arm subsystem are used twice for the left and the right side of the robot, the other subsystems are used once yielding a total of $N_s = 6$ subsystems for the robot. The free floating body subsystem is the only root of the kinematic topology of the robot, and all other subsystems are attached to the subsystem. All of the above mentioned subsystems consist of structural elements and actuators, which themselves can be grouped into smaller subsystems. Thus we start with the smallest subsystem consisting of a motor, a gear, and a structural element attached to the driven side of the gear (subsequently referred to as the drive subsystem) as shown in the left part of Fig. 3, and compute the subsystem matrices and vectors. Next we adopt the above subsystem with the inertial parameters of the individual drive subsystems of the considered limb and then recursively combine these drive subsystems to the limb subsystem (see Fig. 3 in the middle). Then the equations of motion of the complete robot are synthesized by recursively combining the limb subsystems using Eq. 11. As a result we find the equations of motion of the humanoid robot system with free moving base as given in Eq. 7.

### 2.3 Drive Subsystem

The drive subsystem is considered the smallest union of structural elements and an actuator. A schematic of the subsystem can be seen in Fig. 4. A drive subsystem consists of a motor, an elastic gear and an arm attached to the drive side of the gear. The subsystem has two internal degrees of freedom, the position of the motor and the position of the arm. The arm position is given by $q_{A,j}$, the motor position transferred to the gear output is given by $q_{M,j}$. The gear elasticity is modeled as a rotational spring at the output of the gear. While in Eq. 9 one can use a different arbitrary reference frame $R$ for every body of the subsystem, in most cases it is constructive to use a body fixed frame $\mathcal{A}_i$ attached to the arm as a reference frame, as in this case the inertia tensor of a rigid
and arm position \( q \)

External forces, such as motor torques, gravity or contact forces are taken into account by the subsystem matrices and vectors. Again reaction forces of the robot, in the subsystem equations we still have to consider reaction forces of the neighboring joint units. The various drive subsystems are combined to form the subsystem equations.

Using the generalized subsystem velocities \( \dot{y}_{d,j} = [v^T_{F,j} \ \dot{\omega}_{F,j} \ \dot{q}^T_{F,j}]^T \) with the vector of motor and arm position \( q_{j,j} = [q_{M,j} \ q_{A,j}]^T \) \( \in \mathcal{J}_j \) and evaluating the terms in brackets in Eq. 9 yields the subsystem matrices and vectors for the drive subsystem. In contrast to the equations of motion of the robot, in the subsystem equations we still have to consider reaction forces \( Q'_{d,j} \) introduced by the neighboring joint units.

\[
M_{d,j} \ddot{y}_{d,j} + G_{d,j} \dot{y}_{d,j} - Q'_{d,j} = 0.
\] (12)

### 2.4 Limb Subsystem

Each limb subsystem consists of \( N_l \) drive subsystems. The generalized velocities of a limb subsystem are chosen as \( \dot{y}_{l,i} = [v^T_{l,i} \ q^T_{l,i}]^T \) where \( v^T_{l,i} \) is the twist of the origin of the body fixed frame \( \mathcal{L}_i \) attached to the torso. The vector \( q_{j,j} \subset \mathcal{J} \) combines all internal degrees of freedom of the particular limb subsystem. The various drive subsystems are combined to

\[
\begin{bmatrix}
\frac{\partial y_{d,1}}{\partial y_{l,i}}^T \\
\vdots \\
\frac{\partial y_{d,N_l}}{\partial y_{l,i}}^T
\end{bmatrix}
\begin{bmatrix}
M_{d,1} \ddot{y}_{d,1} + G_{d,1} \dot{y}_{d,1} - Q'_{d,1} \\
\vdots \\
M_{d,N_l} \ddot{y}_{d,N_l} + G_{d,N_l} \dot{y}_{d,N_l} - Q'_{d,N_l}
\end{bmatrix}
= Q'_{l,i}
\] (13)

and yield the subsystem equations \( M_{l,i} \ddot{y}_{l,i} + G_{l,i} \dot{y}_{l,i} - Q'_{l,i} = 0 \) with the limb subsystem matrices and vectors. Again reaction forces \( Q'_{l,i} \) have to be considered on the right hand side of Eq. 13.

### 2.5 External Forces

External forces, such as motor torques, gravity or contact forces are taken into account by the generalized forces \( Q \) in the Eq. 7. The motor torques and the gravity forces are considered best
during the assembly of the different subsystems by $R M^r$ respectively $R f^r$ in Eq. 9. The ground contact forces are inserted by means of the principal of virtual work afterward. The generalized forces $Q = Q_c + Q_r$ can be separated into one part representing the contact forces $Q_c$ and another one for the remaining forces $Q_r$. The vector $Q_c$ is calculated as

$$Q_c = \sum_j \kappa_j \left( \frac{\partial V^b_j}{\partial \dot{s}} \right)^T W_j$$

with $V^b_j$ as the body-twist of a reference frame $F_j$ fixed within the $j$-th contact area and $W_j$ as the contact wrench acting in the reference frame. The homogenous transformation of the reference frame w.r.t. the world fixed frame is given by $H_j = (r_j, R_j)$. Contacts can be opened and closed by the scalars $\kappa_j$ which are equal to one if the contact is active or zero otherwise. Due to the unilateral contact between the robot and the contact areas, the contact wrenches $W_j$ have to be limited. Aligning the $z$-axis of the frame $F_j$ with the surface normal and placing it in the middle of the corresponding contact area simplifies the definition of this constraints. For a rectangular shaped contact area, like a foot, the constraints can be expressed by

$$0 \leq e_i^T W_j$$

$$-\frac{b}{2} e_i^T W_j \leq e_i^T W_j \leq \frac{b}{2} e_i^T W_j$$

$$-\frac{a}{2} e_i^T W_j \leq e_i^T W_j \leq \frac{a}{2} e_i^T W_j,$$

with $e_i$ as the $i$-th unit vector and $a$ and $b$ as the length and the width of the foot. As one can see, the perpendicular contact force ($e_i^T W_j$) must always be positive, as the various contact areas can only push on the ground. The limitation of the horizontal torques ($e_4^T W_j$ and $e_5^T W_j$) are a function of the vertical contact forces and the geometry of the foot.

2.6 Equations of Motion

After one has found the subsystem matrices and vectors for the various limb subsystems, they can now be combined to the equations of motion, given by Eq. 7, of a robot with free floating base by using again Eq. 11. The generalized velocities $\dot{s}$ are used as defined in Eq. 2. As the generalized coordinates we use $q \in Q$. The equations of motion are finally given by

$$M(q) \ddot{s} + h(\dot{s}, q) = \begin{bmatrix} 0 \\ \mathbf{B} \end{bmatrix} \tau + \sum_j \kappa_j \left[ A \dot{H}_{jB}^T J_{jB}^T \right] W_j$$

where $\tau$ are the joint torques, $h$ summarizes all nonlinear terms and $\mathbf{B}$ is the input mapping matrix. Contact wrenches are mapped into the configuration space by the Jacobians $J_j \in \mathbb{R}^{6 \times (n+6)}$. The transpose adjoint matrix for the homogeneous transformation $H_{jB}$ is given by $A H_{jB}^T$. Considering the unilateral contact between the feet and the ground additionally to Eq. 18 the constraints from Eqs. 15-17 for the contact wrenches have to be considered.

2.7 Equations of Motion w.r.t Center of Mass

For the derivation of the equations of motion an intuitive choice of the generalized velocities is given by Eq. 2. Using the CoM velocity and the inertial torso angular velocity instead of the torso twist gives many benefits during controller design. Switching generalized velocities from Eq. 2 to Eq. 5 is realized by adding the Jacobian $J_1 = \partial \dot{s}/\partial \dot{z}_1$ to Eq. 9 [9] and finally leads to the equations
of motion with CoM velocity and floating base

\[
\begin{bmatrix}
    mI & 0 & 0 \\
    0 & M_{0b}(q_j) & 0 \\
    0 & M'_{0b,j}(q_j) & M_j(q_j)
\end{bmatrix}
\begin{bmatrix}
    \ddot{z}_i \\
    h_{0b}(\dot{z}_i, q_j) \\
    h_j(\dot{z}_i, q_j)
\end{bmatrix}
+ \begin{bmatrix}
    -mg \\
    h_{0b}(\dot{z}_i, q_j) \\
    h_j(\dot{z}_i, q_j)
\end{bmatrix}
= \begin{bmatrix}
    0 \\
    0 \\
    B
\end{bmatrix}
\sum_j \kappa_j
\begin{bmatrix}
    R_j & 0 \\
    J_{0b,j} & J_{j,j}
\end{bmatrix}
W_j.
\]  

(19)

The gravity vector is given by \( \mathbf{g} \in \mathbb{R}^3 \), \( m \) denotes the total weight of the considered system and \( \mathbf{h}_{0b} \) and \( \mathbf{h}_j \) represent the centrifugal and Coriolis forces.

### 3 Reduced Dynamic Models

In the previous section we derived a multi-body model for the robot that considers elastic gears. In particular for the considered robot this leads to 34 degrees of freedom. This model is adequate for complex and accurate simulations of the robot, especially if one is interested in the dynamics of the drives, but for other purposes, like controller design, a reduced model of the robot is necessary. In this section two different reduced models with diminishing complexity are derived. The first one assumes ideal kinematic gears which reduces the model to 20 degrees of freedom. The last one assumes high gain joint controllers and therefore neglects the joint dynamics and leads to 6 degrees of freedom. While the last two models are used for controller design, the first model can be used for simple simulations to check the stability of the robot.

#### 3.1 Ideal Kinematic Gears

For an ideal kinematic gear the velocity of the attached components corresponds to the velocity of the motor, therefore the constraint \( \Phi = (\dot{q}_{A,1} - \dot{q}_{M,1} \ldots \dot{q}_{A,n} - \dot{q}_{M,n})^T = 0 \) and Eq. 7 yields

\[
M \dot{s} + G(q, \dot{s}) \dot{s} - Q(q) - \left( \frac{\partial \Phi}{\partial s} \right)^T \lambda = 0.
\]  

(20)

Using the orthogonality relation (see [1]) results in

\[
\left( \frac{\partial \dot{s}}{\partial s} \right)^T [M \dot{s} + G(q, \dot{s}) \dot{s} - Q(q)] = 0
\]  

(21)

where \( \dot{s} \) conveniently is chosen as a subset of \( s \) (e.g. by removing the motor velocity).

#### 3.2 High-Gain Joint Controller

The joint angles \( q_j \) can easily be stabilized by a simple high-gain position control law. Doing so and using the singular perturbation theory it is possible to reduce the dynamical model to

\[
\begin{bmatrix}
    mI & 0 \\
    0 & M_{0b}
\end{bmatrix}
\begin{bmatrix}
    \dot{\varepsilon}_c \\
    \dot{\omega}_B
\end{bmatrix}
+ \begin{bmatrix}
    -mg \\
    M_{0b} \ddot{q}_d + \mathbf{h}_{0b}
\end{bmatrix}
= \sum_j \kappa_j
\begin{bmatrix}
    I & 0 \\
    \dot{r}_{c,j} & I
\end{bmatrix}
\begin{bmatrix}
    \mathbf{f}_j \\
    \tau_j
\end{bmatrix}.
\]  

(22)

where \( \mathbf{f}_j \) and \( \tau_j \) are the contact forces and torques with respect to \( \mathcal{W} \) corresponding to the contact wrench \( \mathbf{W}_j \). The vector from the CoM to the center of the contact area is given by \( \mathbf{r}_{c,j} = \mathbf{r}_j - \mathbf{r}_c \). While the dynamical model from Eq. 22 still considers the full multi-body dynamics one has to notice that the reduced model assumes perfect joint tracking \( (q_j - q_j \rightarrow 0) \). The superscript \( c \) in Eq. 22 marks desired values. Using the relation for the time derivative of the inertia matrix \( \mathbf{M} = \mathbf{G} + \mathbf{G}^T \) the Eq. 22 can be rewritten in terms of canonical momenta as

\[
\begin{bmatrix}
    \mathbf{p} \\
    \mathbf{L}
\end{bmatrix}
+ \begin{bmatrix}
    -mg \\
    0
\end{bmatrix}
= \sum_j \kappa_j
\begin{bmatrix}
    I & 0 \\
    \dot{r}_{c,j} & I
\end{bmatrix}
\begin{bmatrix}
    \mathbf{f}_j \\
    \tau_j
\end{bmatrix}.
\]  

(23)

where \( \mathbf{L} \) is the overall angular momentum and \( \mathbf{p} \) is the total linear momentum given by

\[
\begin{bmatrix}
    \mathbf{p} \\
    \mathbf{L}
\end{bmatrix}
= \begin{bmatrix}
    I & 0 & 0 \\
    0 & I & 0
\end{bmatrix} \mathbf{M}(q_j) \dot{s}
\]  

(24)

with \( \mathbf{M}(q_j) \) is the inertia matrix from Eq. 19.
4 CONTROL APPLICATIONS

The derived multi-body model and the various reduced models can be used in a wide range for different control applications. While many walking controllers use a pointmass model for feedback control \[8, 3\], only a few controllers consider the angular momentum of the robot \[5, 3, 7\]. Also for walking pattern generation most robots rely on the pointmass model, as in this case an online generation of the CoM trajectory can be realized \[4, 11\]. Considering the angular momentum during walking pattern generation can either be realized offline with the full robot dynamics given by Eq. 19 \[2\] or online by estimating the angular momentum \[6\] through equivalent masses e.g. in the legs, or by the model from Eq. 22. It is clear that depending on the control application a different one of the derived models fits best. Two different control applications, one using the model from Eq. 22 and another one using the dynamics of the multi-body model from Eq. 18 will demonstrate how the different models can be used for the control tasks.

4.1 Feed-Forward Torques

In the transition from Eq. 19 to Eq. 22 we assumed a high gain position control law and sufficiently accurate joint position tracking. On the physical robot the accuracy of joint tracking can be significantly increased by means of torque feed-forward control. To obtain feed-forward joint torques for a given trajectory in the configuration space \((q, \dot{s}, \ddot{s})\) one needs to calculate the inverse dynamics from Eq. 18. During single support the contact wrench acting on the swing leg is equal to zero, which leads to a unique solution of the inverse dynamics. In contrast the system is overactuated during double support leading to an infinite number of solutions for the inverse dynamics. To find a solution during double support one needs to find a criterion, for example the minimum of the contact forces, to obtain a favorable solution. Solving a quadratic optimization problem of the form

\[
\min_{W_L, W_R} \alpha W_L^T \Omega W_L + (1 - \alpha) W_R^T \Omega W_R
\]

s.t.
\[
\kappa_L A_{H_{Lb}}^T W_L + \kappa_R A_{H_{Rb}}^T W_R = W,
\]

with \(\alpha \in [0, 1]\) as a scalar coefficient weighting the contact forces to allow a smooth transition between single and double support and \(\Omega\) as diagonal scaling matrix leading to a dimensionless objective function, gives the optimal force distribution that minimizes the Euclidian norm of the contact forces. The vector \(W\) represents the wrench that has to act at the upper-body \(B\) to realize the desired motion and is given by the first six equations of the left hand side of Eq. 18.

\[
M(q) \ddot{s} + h(\dot{s}, q) = \begin{pmatrix} W \\ \tau \end{pmatrix}.
\]

As already mentioned there is only a unilateral contact between a foot and the ground, which limits the set of feasible contact wrenches and the constraints from Eqs. 15-17 need to be considered. By limiting \(e_i^T W_k, e_i^T W_k, e_i^T W_k\) with \(k \in \{L, R\}\) and \(e_i\) as the \(i\)-th unit vector, according to a physical friction law, slipping between the ground and the feet can be prevented as further constraints. In this work sufficient friction is assumed which has held for all tested walking trajectories in simulations and in real world so far. According to Eq. 18 the desired joint torques \(\tau_f\) can now be calculated from the distributed ground reaction forces. If one would just use \(\kappa_L\) and \(\kappa_R\) to turn the contacts on and off during single and double support this would lead to non-continuous ground reaction forces and therefore to non-continuous joint torques. This can be omitted by choosing an appropriate distribution coefficient \(\alpha\). A good choice can be made by the relative position of the zero-moment point relative to the origins of the feet. The position of the zero-moment point can be obtained by transforming \(W\) into \(\omega\) and then by dividing the resulting torque by the normal force. While Fig. 5 gives an idea of how \(\alpha\) can be chosen, any other rule for the force distribution will lead to similar results. Figure 6 exemplary shows the desired contact forces and torques for a typical walking gait where the robot walks forward.
4.2 Angular Momentum Control

In the walking pattern generation typically reference trajectories for the lower body coordinates are designed according to the dynamic constraints of the unilateral contact of the feet to the ground. While in most cases only the overall linear momentum $p$ from Eq. (23) of the robot is considered during this trajectory design stage, the neglected angular momentum $L$ can cause the robot to slip and rotate about its vertical axis[5]. Using the arms, like humans do during walking, to compensate the angular momentum caused by the joints of the lower body can reduce the overall angular momentum significantly.

The total angular and linear momentum of the robot is given by Eq. 24 where the inertia matrix gives the relation between the total momentum and the generalized velocities.

While the linear momentum $p$ is a linear function of the CoM acceleration, the overall angular momentum $L$ is a function of the joint velocities.

A short excursion to inverse kinematics calculations shows that the inverse kinematics for a redundant robot can be calculated by using the differential kinematics Jacobian[10]

$$\dot{z}_i = J_i \dot{s}$$

(28)

with $z_i$ from Eqs. 3-4 and their corresponding Jacobians. The velocities in the configuration space can be obtained by

$$\ddot{s} = J_i^{-1} \dot{z}_i.$$  

(29)

Knowing the initial configuration of the robot, velocities can be summed up over time to get the corresponding configuration $x_i$ of the robot.

Instead of using the generalized velocities $z_i$ in Eq. 29 we introduce a new vector

$$\dot{z}_m = S \begin{bmatrix} z_l \\ L \end{bmatrix} = S \begin{bmatrix} J_l & 0 \\ I & 0 \end{bmatrix} M \ddot{s}$$

(30)

which is an extension of the generalized velocities $z_l$ by the total angular momentum. The matrix $S \in \mathbb{N}^{(6+n) \times (12+n)}$ is a binary selection matrix used to pick the $6+n$ new coordinates of interest. If we want to use the arms ($q_{13}$ and $q_{14}$) of the robot in Fig. 3, with only one degree of freedom per arm, to compensate for the angular momentum around the vertical axis we can use the new vector $\dot{z}_m = (\dot{r}_C \, \alpha_B \, V_{L}^b \, V_{R}^b \, \dot{q}_{13} + \dot{q}_{14} \, L_c)^T$. Instead of directly choosing the velocity of the second arm we introduce $\dot{q}_{13} + \dot{q}_{14}$ as relative velocity of the two arms to tie their motion together.
Angular Momentum
roll in Nms

Angular Momentum
pitch in Nms

Angular Momentum
yaw in Nms
time in s

Figure 7. Time evolution of the angular momentum for a typical walking gait: (left) without angular momentum compensation, (right) with angular momentum compensation.

For the control of biped robots in general we are only interested in the joint angles \( q^J \) for a given trajectory \( x_2 \) in the configuration space \( \mathcal{Z}_2 \). Thus instead of Eq. 28 we can also use

\[
\dot{z}_m = SJ_mJ_1^{-1}\dot{z}_1
\]

\[
= \begin{bmatrix}
I & 0 & 0 & 0 & 0 \\
0 & I & 0 & 0 & 0 \\
0 & 0 & I & 0 & 0 \\
0 & 0 & 0 & I & 0 \\
0 & 0 & 0 & 0 & e_1 + e_2 \\
0 & 0 & 0 & 0 & e_3
\end{bmatrix}
\begin{bmatrix}
0 \\
0 \\
A_1 \\
A_2 \\
A_3 \\
A_4 \\
A_5 \\
A_6 \\
0 \\
e_{13} + e_{14} \\
e_{13}M_{0b} + e_{14}M_{0b,J}
\end{bmatrix}
\]

\[
J_mJ_1^{-1} = \begin{bmatrix}
I & 0 & 0 \\
0 & I & 0 \\
A_1 & A_2 & A_3 \\
A_4 & A_5 & A_6 \\
0 & 0 & e_{13} + e_{14} \\
0 & e_{13}M_{0b} + e_{14}M_{0b,J}
\end{bmatrix}
\]

to calculate the inverse kinematics. Adopting the special structure of the Jacobian in Eq. 32 we get

\[
\dot{\mathbf{q}}_J = \begin{bmatrix}
A_3 \\
A_6 \\
e_{13} + e_{14} \\
e_{13}M_{0b,J}
\end{bmatrix}^{-1}
\begin{bmatrix}
V_R^b \\
V_L^b \\
\dot{q}_{13} + \dot{q}_{14} \\
L_z
\end{bmatrix}
\begin{bmatrix}
\mathbf{r}_c \\
\mathbf{r}_c \\
\mathbf{r}_c \\
\mathbf{r}_c
\end{bmatrix}
\begin{bmatrix}
A_1 \\
A_2 \\
A_3 \\
A_4 \\
A_5 \\
A_6 \end{bmatrix}
\begin{bmatrix}
0 \\
0 \\
e_{13} + e_{14} \\
e_{13}M_{0b} + e_{14}M_{0b,J}
\end{bmatrix}
\]

The joint velocities are then given by

\[
\dot{\mathbf{q}}_J = \begin{bmatrix}
A_3 \\
A_6 \\
e_{13} + e_{14} \\
e_{13}M_{0b,J}
\end{bmatrix}^{-1}
\begin{bmatrix}
V_R^b \\
V_L^b \\
\dot{q}_{13} + \dot{q}_{14} \\
L_z
\end{bmatrix}
\begin{bmatrix}
A_1 \\
A_2 \\
A_3 \\
A_4 \\
A_5 \\
A_6 \end{bmatrix}
\begin{bmatrix}
0 \\
0 \\
e_{13} + e_{14} \\
e_{13}M_{0b} + e_{14}M_{0b,J}
\end{bmatrix}
\]

(34)

With an Euler integration method joint angles can be calculated to \( \mathbf{q}_J(t_{k+1}) = \mathbf{q}_J(t_k) + \dot{\mathbf{q}}_J(t_k)\Delta t \).

To avoid a numerical drift a stabilization term needs to be added to the general formulation of the inverse kinematics given by Eq. 34. A more detailed analysis of the numerical stabilization of the inverse kinematics with the geometric Jacobian for different orientation representations can be found in [10].

In Fig. 7 one can see the time evolution of the angular momentum generated once with a classical inverse kinematics and once generated with the proposed angular momentum control. It can be
Figure 8. Time evolution of the vertical torque around the CoM measured by the force/torque sensors in the feet: (top) without angular momentum compensation, (bottom) with angular momentum compensation.

Figure 9. Snapshots of the robot walking: (top) without angular momentum compensation, (bottom) with angular momentum compensation.

seen that the vertical angular momentum is constant at zero if the compensation is turned on. While in Fig. 7 we can only see the feed-forward controlled angular momentum, in Fig. 8 we can see the torque around the CoM for a simulated robot. It can be seen that the torque around the CoM can be reduced significantly by the proposed angular momentum compensation.

5 CONCLUSIONS

In this paper we presented a detailed dynamical model that can be used for the control of humanoid robots. The derivation of the dynamical model uses various subsystems combined recursively to the equations of motion. The used methods allow to easily adopt the kinematic structure of single limbs and to reuse results obtained for limbs with similar kinematic structures but different inertial parameters such as for symmetric legs. After we found a recursive formulation to calculate the equations of motion we perform various state transformations and apply some model simplifications to gain equations that can be used to solve control problems effectively. Next to the modeling of the robot we presented two control applications that use the derived model simplifications. For feed-forward torque control inverse dynamics is used together with a parameter optimization to calculate necessary joint torques. Further on we extended the inverse kinematics algorithm of the robot to resolve a desired momenta to a motion of the robot. Simulations showed the effectiveness of the proposed algorithm.

ACKNOWLEDGEMENT

This work has been supported by the Austrian COMET-K2 program of the Linz Center of Mechatronics (LCM), and was funded by the Austrian federal government and the federal state of Upper Austria.
REFERENCES


Modelling and Control Synthesis of Flexible Robot Arm Equipped with Additional Sensors

Šika Zbyněk*, Volech Jiří, Mráz Ladislav, Valášek Michael, Beneš Petr

*Faculty of Mechanical Engineering
Czech Technical University in
Prague
Technická 4, Praha 6
166 07, Czech Republic
Zbynek.Sika@fs.cvut.cz

ABSTRACT

The work deals with the introduction of fundamental problems of improving the accuracy of the robots with serial kinematic structure that can perform various operations such as machining. For the control synthesis is necessary to create a good model of flexible multibody system, which is outlined in the article. The robot is equipped with an auxiliary measuring apparatus for measuring the deformations and the position of the individual parts. The design of advanced robot control is realized based on the model with additional measurements. The first step of the control design is realized by the cascade control of particular drives based on the inverse kinematics. The tuned cascade control is used as the initial estimate for the optimization of the structured fixed order H-infinity robust control.

Keywords: Flexible multibody system, cascade control, H-infinity, redundant measurement.

1 INTRODUCTION

The machining with the industrial robot of a serial structure is becoming nowadays the interesting field of the research and development. The serial structure of the industrial robot has wider range of achievable operating space. Their movement ability and the lower cost than conventional machining tools are undoubted advantage. This advantage is compensated by the significantly lower accuracy caused by several factors. These factors include: greater flexibility of the robot arms, large heat deformations, gearboxes compliances (which represent the main sources of the robot compliance), vibrations etc. There are several investigated approaches of how to achieve such a complicated goal of the accuracy enhancement. One of the approaches is the force feedback [1]. This approach deals with the measurement of the direct contact force and their compensation. The main problem of this approach is the correct analysis of the force signal. Other approaches are presented in project Advocut [2]. For the gearboxes deformations compensation the driven site of the gears were equipped with the rotational position sensors. This approach deals with the problem of overall compliance compensation by controlling speed of each gearbox separately. Therefore the controller influences only a part of the compliance [3]. The analytical prediction of the deformation of the robotic structure is another approach described in [4], [5]. Another project Comet [5], [6], [7] has a good result. This project is based on the full FEM model of the robot and prediction of the contact forces. The workpiece is mounted on the robot arm and carries out main displacements while small inaccuracies are compensated by the machining tool which is attached to a piezo actuated table. This good result is balanced by the significant reduction of the workspace and reduces the mobility advantage of the serial robot. To achieve similar or better results, with smaller restriction on the workspace, different ideas are necessary. In this case the on-line measurements of the robot end-effector together with the overall mathematical model will be developed. Afterwards the actual deviation are going to be compensated by the feedback strategies.

Several possible concepts of the redundant measurements of the flexible arm and its end-effector motion using the additional auxiliary mechanisms and/or laser beam sensors have been analyzed.
by authors in [8]. The chosen additional set of sensors is shown in the Figure 1. Based on this previous research it has been currently developed the control of the entire mechanism. The first part of this research is presented in the paper.

Figure 1. Considered concept of additional measurements

2 FLEXIBLE MULTIBODY MODEL WITH SENSORS

2.1 Flexible multibody model

The robot Mitsubishi RV-6S (Figure 2.) has been chosen as the sample for the investigation of the control concept. The robot has six degrees of freedom and therefore it is suitable for the task of various applications. It consists of seven parts (including the fixed base), but only the longest arms are considered as a flexible (arm 3 and 5). All gearboxes are also considered as flexible. Two sets of coordinates have been used for the description of the whole robot position. The traditional joint coordinates are used for the movement description of the joints: \( \mathbf{q} = [q_1, q_2, q_3, q_4, q_5, q_6]^T \) and the modal coordinates describe the arms deformations: \( \mathbf{E} = [e_3, e_5] \).

Figure 2. Structure of the robot RC-6S model

The modal coordinates express the deformation (bending and torsional) of the parts in an arbitrary point \( P \) (Figure 3.) as the function of the corresponding eigenmodes. The translational deformation in point \( P \) is
\[ \mathbf{u}_{li}^{p} = \begin{bmatrix} u_x(x_p, y_p, z_p, t) \\ u_y(x_p, y_p, z_p, t) \\ u_z(x_p, y_p, z_p, t) \end{bmatrix}_i = \begin{bmatrix} \quad \dot{\varphi} x_1(x_p, y_p, z_p) \\ \quad \dot{\varphi} y_1(x_p, y_p, z_p) \\ \quad \dot{\varphi} z_1(x_p, y_p, z_p) \end{bmatrix}_i \begin{bmatrix} \begin{bmatrix} \dot{e}_1(t) \\ \dot{e}_2(t) \\ \vdots \end{bmatrix} \end{bmatrix}. \tag{2.1} \]

This expression can be simplified by
\[ \mathbf{u}_{li}^{p} = \begin{bmatrix} \dot{\varphi}_1 \mathbf{e}_i \end{bmatrix}. \tag{2.2} \]

The matrix \( \dot{\varphi}_i \) from (2.2) is constant and the whole time dependence of the deformation is described by the vector of the modal coordinates \( \mathbf{e}_i \). The number of its elements corresponds to the number of eigenmodes being used for the deformation description. Analogously, the influence of the torsional deformation in the point \( P \) can be derived as
\[ \mathbf{\varphi}_{li}^{p} = \begin{bmatrix} \varphi_x(x_p, y_p, z_p, t) \\ \varphi_y(x_p, y_p, z_p, t) \\ \varphi_z(x_p, y_p, z_p, t) \end{bmatrix}_i = \begin{bmatrix} \quad \varrho x_1(x_p, y_p, z_p) \\ \quad \varrho y_1(x_p, y_p, z_p) \\ \quad \varrho z_1(x_p, y_p, z_p) \end{bmatrix}_i \begin{bmatrix} \begin{bmatrix} \varrho_1(t) \\ \varrho_2(t) \\ \vdots \end{bmatrix} \end{bmatrix}. \tag{2.3} \]

This expression can be simplified by
\[ \mathbf{\varphi}_{li}^{p} = \begin{bmatrix} \varrho \mathbf{e}_i \end{bmatrix}. \tag{2.4} \]

The meaning of the vectors and matrices describing the rotational deformation in (2.3) and (2.4) is similar to the previous case of the translational deformation. The flexible links are considered as the grid of points in which these linear combinations of the modal-shapes can be assembled and the deformation can be thus obtained in these points using the modal coordinates. As stated before, the matrices of the eigenmodes \( \dot{\varphi}_i \) and \( \varrho \) are constant and their elements can be obtained using the FEM analyses of the particular robot parts. The main advantage of the usage of the modal coordinates \( \mathbf{e}_i \) is the reduction of the states of the complex system. It is not necessary to describe the Cartesian coordinates (and their velocities and accelerations) of each point in the grid representing the flexible body. The reduced number of eigenmodes is sufficient for the description of the deformation. The dynamics of the deformation is given by equations (2.1)-(2.4) and their time derivatives.

![Figure 3. Deformation of the flexible body](image)

The last set of coordinates represent the position of the electrical drive rotors: \( \mathbf{q}_M = [q_{1M}, q_{2M}, q_{3M}, q_{4M}, q_{5M}, q_{6M}]^T \). The connection to the arms is realized by the flexible gearboxes with corresponding stiffness, as mentioned before. Using the recursive formalism for the flexible bodies [10] the final system of equations of motion (EOM) can be obtained.
The symbol $M_R$ represents the part of the mass matrix related to the „rigid“ motion, $M_F$ is the part of the mass matrix representing the „flexible“ motion and $M_{RF}$ represents the interconnection of them. The symbol $M_M$ represents the mass matrix of the rotors. The symbols on the right-hand side are the corresponding generalized force vectors.

With the notation $y = [q, \dot{q}, q_M]^T$, the system (2.5) can be simplify to

$$M_y \ddot{y} = Q_y.$$  \hspace{1cm} (2.6)

Denoting $x_1 = y$ and $x_2 = \dot{y}$ the system of differential equations of the first order is

$$\dot{X} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ M_y^{-1} Q_y \end{bmatrix}.$$  \hspace{1cm} (2.7)

The model of robot dynamics based on (2.7) has been assembled in the Matlab-Simulink environment. The parameters of the model had to be estimated according to material of the robot parts and their dimensions. The eigenmodes of the flexible arms have been obtained using the FEM analysis. The currently developed functional model will be identified in order to make the mathematical model more accurate.

### 2.2 Measurement of arm deformations

The additional set of sensors along the robot structure has been considered for the deformation detection. The source of the laser beam is mounted on one side of the arm and the sensor for detection of the laser spot position on the second one. There are several possibilities to detect the laser spot. The four-quadrant-sensor (4QD) (Figure 4) has been chosen, taken into account expected values of deformations less than 50 µm.

![Figure 4. Four-quadrant-detector](image)

The 4QD detector consists of the four photodiodes A, B, C, D. These diodes have the equal areas and are separated by the small gaps. Photodiodes transform incoming laser beam to the currents $I_a$, $I_b$, $I_c$, $I_d$, afterwards the currents are transformed in the operational amplifier to the relative voltage levels $V_a$, $V_b$, $V_c$, $V_d$. Voltage level generated by the photodiode is proportional to the optical energy illuminating its surface. The evaluation of the spot position on the 4QD assumes that the laser spot has perfectly circular shape with uniform distribution of power. Concerning this assumption, when the laser beam falls into the perfect center $x=y=0$, the generated voltages from each quadrant are equal. The spot displacement along the x-axis or y-axis of the detector
will be detected as a relative change between these four current outputs \[11\]. These currents are calculated in following manner in order to calculate the so called pointing error, relative to the center of the detector

\[
d_x = K_x \frac{(I_a+I_d)-(I_b+I_c)}{I_a+I_b+I_c+I_d} = K_x \frac{(V_a+V_d)-(V_b+V_c)}{V_a+V_b+V_c+V_d},
\]

(2.8)

\[
d_y = K_y \frac{(I_a+I_b)-(I_c+I_d)}{I_a+I_b+I_c+I_d} = K_y \frac{(V_a+V_b)-(V_c+V_d)}{V_a+V_b+V_c+V_d}.
\]

(2.9)

\(K_x\) and \(K_y\) are the correlation coefficients of the x- and y-axis directions respectively. Two main domains will be considered. The first part is the shape of the laser spot (it is assumed the perfect circular shape) and second is the energy input to the illuminated area of each photodiode from the laser beam (it is assumed the uniform distribution of the light energy) \[12\]. Then the energies on each illuminated quadrant are proportional to the corresponding areas

\[
S_A = \frac{\pi r^2}{4} + xy + \frac{x}{2} \sqrt{r^2 - x^2} + \frac{y}{2} \sqrt{r^2 - y^2} + \frac{r^2}{2} \left[ \sin^{-1} \left( \frac{x}{r} \right) + \sin^{-1} \left( \frac{y}{r} \right) \right]
\]

(2.10)

\[
S_B = \frac{\pi r^2}{4} - xy - \frac{x}{2} \sqrt{r^2 - x^2} + \frac{y}{2} \sqrt{r^2 - y^2} - \frac{r^2}{2} \left[ \sin^{-1} \left( \frac{x}{r} \right) + \sin^{-1} \left( \frac{y}{r} \right) \right]
\]

(2.11)

\[
S_C = \frac{\pi r^2}{4} + xy - \frac{x}{2} \sqrt{r^2 - x^2} - \frac{y}{2} \sqrt{r^2 - y^2} - \frac{r^2}{2} \left[ \sin^{-1} \left( \frac{x}{r} \right) + \sin^{-1} \left( \frac{y}{r} \right) \right]
\]

(2.12)

\[
S_D = \frac{\pi r^2}{4} - xy + \frac{x}{2} \sqrt{r^2 - x^2} - \frac{y}{2} \sqrt{r^2 - y^2} + \frac{r^2}{2} \left[ \sin^{-1} \left( \frac{x}{r} \right) + \sin^{-1} \left( \frac{y}{r} \right) \right]
\]

(2.13)

Where \(x\) and \(y\) are the displacements of the spot center respectively to the center of the 4QD sensor and \(r\) is the radius of the laser spot. Therefore the pointing errors \(d_x\) and \(d_y\) can be also calculated using the illuminated areas of the quadrants

\[
d_x = K_x \frac{(S_A+S_d)-(S_b+S_c)}{S_a+S_b+S_c+S_d} \frac{1}{\pi r^2} \left[ 2xy\sqrt{r^2 - x^2} + 2r^2 \sin^{-1} \left( \frac{x}{r} \right) \right]
\]

(2.14)

\[
d_y = K_y \frac{(S_a+S_b)-(S_c+S_d)}{S_a+S_b+S_c+S_d} \frac{1}{\pi r^2} \left[ 2xy\sqrt{r^2 - y^2} + 2r^2 \sin^{-1} \left( \frac{y}{r} \right) \right]
\]

(2.15)

Substituting (2.10)-(2.13) to the (2.14) and (2.15) the mathematical model for the 4QD sensor pointing errors and the displacement are as follows

\[
d_x = K_x \frac{(S_a+S_d)-(S_b+S_c)}{S_a+S_b+S_c+S_d} \frac{1}{\pi r^2} \left[ 2x\sqrt{r^2 - x^2} + 2r^2 \sin^{-1} \left( \frac{x}{r} \right) \right]
\]

(2.16)

\[
d_y = K_y \frac{(S_a+S_b)-(S_c+S_d)}{S_a+S_b+S_c+S_d} \frac{1}{\pi r^2} \left[ 2y\sqrt{r^2 - y^2} + 2r^2 \sin^{-1} \left( \frac{y}{r} \right) \right]
\]

(2.17)

\[3\] CONTROL SYNTHESIS

The sufficient control design for the whole flexible multibody system is not a trivial task. Several steps of synthesis have been proposed to develop a motion control of the end-effector on the selected trajectories. The first step was the synthesis of a cascade regulators \[13\] for particular robot axis (joints) with the requested joint motion evaluated from the inverse kinematic transformation. The second step uses the optimization of \(H_\infty\) MIMO controller of the fixed order and is described within this paragraph. The third step uses the combination of \(H_\infty\) controller and the computed torque method. The controller synthesis is based on the linearized models in several different arm positions (Figure 5). The regulators obtained from the first step (cascade regulators) has been used as the starting point of the \(H_\infty\) controller optimization.
The state-space representation obtained after linearization is as follows

\[ \dot{x} = Ax + Bu \]
\[ y = Cx + Du. \]

Where \( x \) is the state vector, \( y \) is the output vector and \( u \) is the input (or control) vector. The eigenfrequencies and eigenmodes have been analyzed for considered robot positions (Figure 5). The six highest frequencies correspond to the flexible gearboxes and have approximately the same values. The 8 following lower frequencies, corresponding to the flexible modes of the robot chain, varies among different positions only in the range of 30 %. The state space model of this complex flexible model has relatively large dimensions. Consequently the optimization of the control law is not a simple task because of many optimization parameters. As mentioned above the results of the control design from the previous step (cascade regulators) have been used as the initial guess for this optimization.

### 3.1 Cascade control

The cascade PID controller has been designed in the first phase [13]. These controllers have been tuned for each axis separately and then they have been collected to the controller of the whole robot using the inverse kinematic transformation. The PI regulator has been applied for the regulation of the velocity of the motor. The signal for the regulator has been obtained as the resulting difference between the real and the desired velocity of the motor. Further, the resulting position has been controlled by the simple P regulator (Figure 6.).
Two types of sensors have been used for the control feedback, the motor velocity has been measured by the rotor position encoder, whereas the feedback to the position loop comes from the encoder mounted directly to the particular robot joint. The link deformation sensors haven’t been used for this first level of control. The signal from the P regulator has been filtered by the notch filter in order to delete the first frequencies presented in the signal (10-12 Hz). Although the filter works well and alleviate the low-frequencies, they are not fully suppressed. The controller sufficiently stabilize the system, although the response time has been insufficient.

3.2 $H_\infty$ controller optimized by HIFOO

The second approach uses HIFOO (H-infinity Fixed Order Optimization) to find a stabilizing controller. HIFOO is a public-domain Matlab package that aim on designing a stabilizing linear controller of fixed-order for a linear plant described by generalized state-space form, while minimizing the $H_\infty$ norm of the closed-loop transfer function [14, 15]. The system has the capability to optimize one regulator for the set of several linear systems, for example coming from the linearization of nonlinear system in different positions. The result of optimization of such extensive system like this (46 states) never guarantee to be the best global solution. Quasi-Newton algorithm (BFGS), which is used in the initial phase optimization, is designed to find the local minimum. Other methods (local bundle phase and gradient sampling) control the accuracy of the minimum being reached. It is therefore appropriate to have the reasonable initial estimate, in our case coming from the cascade regulators.

It is necessary to create an appropriate state-space system for the suitable control design. The considered outputs are as follows.

Each link with gearbox is represented by four states:

- Angular position of robot link $g_i$
- Angular velocity of robot link $\dot{g}_i$
- Angular position of motor $g_{mi}$
- Angular velocity of motor $\dot{g}_{mi}$

Where $i$ is the index of robot links $i \in <1:6>$.

Flexibility of arms is represented by modal states:

- four states for position $E_i$ (dim[$E_i$ (*]) = 4x1)
- four states for velocity $\dot{E}_i$ (dim[$\dot{E}_i$ (*]) = 4x1)

Altogether the state vector (dim[$x$(*)] = 40x1) is

$$x = (g_1 \ldots g_6 \ E_3 \ E_5 \ \dot{g}_1 \ldots \dot{g}_6 \ \dot{E}_3 \ \dot{E}_5 \ g_{m1} \ldots g_{m6} \ \dot{g}_{m1} \ldots \dot{g}_{m6})^T. \quad (3.3)$$

The flexibility is considered only for the links 3 and 5.
The sensor output vector (dim[Y(\*)]= 16x1) is

\[ Y = (\dot{g}_1 \ldots \dot{g}_6 \ g_1 \ldots g_6 \ d_{3x} \ d_{3y} \ d_{5x} \ d_{5y})^T. \]  (3.4)

Where measured deformation on the 3\(^{\text{rd}}\) arm in x axe \(d_{3x}\) (2.14) and y axe \(d_{3y}\) (2.15), measured deformation on the 5\(^{\text{th}}\) arm in x axe \(d_{5x}\) (2.14) and y axe \(d_{5y}\) (2.15).

The performance output vector is

\[ Z = (e_x \ e_y \ e_z)^T. \]  (3.5)

Where \(e_x\) corresponds to the difference between the desired and actual position \(R_{\text{ep,x}} - R_{\text{ep,x,actual}}\).

The reference input vector is

\[ W = (R_{\text{ep,x}} \ R_{\text{ep,y}} \ R_{\text{ep,z}})^T. \]  (3.6)

And consists of desired position of the end-point in the based Cartesian coordinates.

Input vector (dim[U(\*)]= 6x1) is

\[ U = (I_1 \ldots I_6)^T. \]  (3.7)

And dimension of the each state space matrices are:

- \(\text{dim}[A(*)]= 40\times40\)
- \(\text{dim}[B(*)]= 40\times6\)
- \(\text{dim}[C(*)]= 16\times40\)
- \(\text{dim}[D(*)]= 16\times6\)

### 3.2.1 Tracking task

The state-space model has to be adapted to the form applicable to HIFFO algorithms [15, 16] in order to create a tracking task of the end-effector of the robot. Following description has to be used in order to hold the HIFOO terminology (Figure 7.): Modified state-space model Plant (P), Controller (K), References (W) and errors (Z).

After this adjustment the extended state-space model (Plant) is obtained.

\[
\begin{bmatrix}
\dot{x} \\
Z \\
Y
\end{bmatrix} =
\begin{bmatrix}
A_1 & B_1 & B_2 \\
C_1 & D_{11} & D_{12} \\
C_2 & D_{21} & D_{22}
\end{bmatrix}
\begin{bmatrix}
x \\
W \\
U
\end{bmatrix} \]  (3.8)
As shown in Figure 7 the state-space model of the system has to be extended to include added vectors of reference inputs and performance outputs. The resulting controller can be in the form of the state-space model of chosen fixed order:

\[
\begin{bmatrix}
\dot{x} \\
U
\end{bmatrix} = \begin{bmatrix}
\hat{A} & \hat{B} \\
\hat{C} & \hat{D}
\end{bmatrix}
\begin{bmatrix}
x \\
Y
\end{bmatrix}
\]  

(3.9)

Vector of inputs \( W \) include a desired position \( R_{\text{ep},x}, R_{\text{ep},y}, R_{\text{ep},z} \) in global coordinates of the robot, rotation of endpoint is not considered at this moment. Only the deviation from the desired position (errors \( e_x, e_y, e_z \)) has been chosen as the evaluation function \( Z \). Low pass filter has to be used on the performance output to eliminate high frequencies (3.8).

\[
\left( \frac{\frac{1}{M}s + Wb}{s + Wb + Aw} \right)^n
\]  

(3.10)

Where \( Wb \) sets the position, \( M \) the upper limit, \( Aw \) lower limit and \( n \) the order of the filter. The results of optimization are very sensitive to the setup of this filter, it is therefore necessary to tune it correctly (figure 8).

Figure 9. Low pass filter

The 0 order stabilizing controller was found by optimization with the initial condition for which the linearized cascade control was used. This controller work after first sets of iteration satisfactorily and stabilized the system correctly, but his response time was unacceptable. Therefore that controller was used as another initial guest for optimization with goal to find the better controller of 1th order. Furthermore, the same procedure was used for finding the suitable controller of the higher order. For the 5th order controller the results did not showed variation greater than 1%. For better results the settings of the filter for deviations could be discussed.

4 TESTING OF CONTROLLER

A simple test has been performed with each controller to evaluate its performance. The test track has been chosen as the Lemniscate of Bernoulli (Figure 9.).
Geometry of the Lemniscate of Bernoulli is a trajectory of the plane curve defined from two given position F1 and F2, known as foci, at distance $2e$ from each other as the locus of points P. Parametric form of the Lemniscate of Bernoulli can be expressed as:

\[
\begin{align*}
x &= \sqrt{2e} \frac{\cos t}{1 + (\sin t)^2} \\
y &= \sqrt{2e} \frac{\sin t \cdot \cos t}{1 + (\sin t)^2}
\end{align*}
\]  \hfill (4.1)

\hfill (4.2)

Results are shown only for the controller of the 5th order. The controller performs good results as shown in Figure 10.
The end-point tracks the trajectory of the Lemniscate well with maximal error of 0.51 mm in x (Figure 11. a) and 0.18 mm in y (Figure 11. b) and 0.1 mm in z (Figure 11. c), although, there was no desired movement in this axis. The further improvement of the control accuracy is the target of the next design step. The improvement of the robot model will be based on the identification of the functional model.

5 CONCLUSIONS

The paper deals with the investigation of the modelling and control of the flexible robot based on the extended set of sensors. Therefore the complex multibody system model of the flexible robot with serial structure has been developed for the simulation experiments. The mathematical model of the four-quadrant-diode has been derived. The model of the robot has been equipped with auxiliary measurement mechanism consisting of the laser beam emitter and the four-quadrant-diode. Several positions of the robot have been analysed and the linearized models for these positions have been used for the robust control design. The previous results of the control design based on the sequence of cascade regulators has been used as the initial guess for the robust controller optimization by the HIFOO algorithm. The optimization procedure considers gradually increasing order of regulators from 0th order to the 5th order. The resulting controller satisfactorily stabilizes the system and significantly improve the positioning accuracy with respect to the cascade regulator solution. Nevertheless the further improvement is necessary in order to fully use the extended set of extended sensors. The further development of control design currently continues using the combination of $H_\infty$ controller and the computed torque method. Moreover the arising functional model will be identified in order to make the multibody model more accurate. The final aim is to verify experimentally the increase of effective stiffness and positioning accuracy of the robotic chain equipped by redundant measurements using position feedback based control.

ACKNOWLEDGEMENTS

The authors appreciate the kind support by the grant GA13-39057S “Position Feedback Based Stiffness Increase of Robots by Redundant Measurement” of the Czech Science Foundation.

REFERENCES


Multi-physics Modelling of a Compliant Humanoid Robot

Alexandra A. Zobova*, Timothee Habra#, Nicolas Van der Noot#, Houman Dallali†, Nikolaos G. Tsagarakis†, Paul Fisette#, Renaud Ronsse#

* Faculty of Mechanics and Mathematics
Lomonosov Moscow State University
Leninskie Gory, 1
119991, Moscow, Russia
azobova@mech.math.msu.su

†Department of Advanced Robotics
Istituto Italiano di Tecnologia
Via Morego, 30, 16163
Genova, Italy
[houman.dallali, nikos.tsagarakis]@iit.it

# Center for Research in Energy and Mechatronics
Université catholique de Louvain (UCL)
Place du Levant 2, 1348
Louvain-la-Neuve, Belgium
[timothee.habra, nicolas.vandernoot, paul.fisette, renaud.ronsse]@uclouvain.be

ABSTRACT

In this paper, we discuss some very important features for getting exploitable simulation results for multibody systems, relying on the example of a humanoid robot. First, we provide a comparison of simulation speed and accuracy for kinematics modeling relying on relative vs. absolute coordinates. This choice is particularly critical for mechanisms with long serial chains (e.g. legs and arms). Compliance in the robot actuation chain is also critical to enhance the robot safety and energy efficiency, but makes the simulator more sensitive to modeling errors. Therefore, our second contribution is to derive the full electro-mechanical model of the inner dynamics of the compliant actuators embedded in our robot. Finally, we report our reasoning for choosing an appropriate contact library. The recommended solution is to couple our simulator with an open-source contact library offering both accurate and fast full-body contact modeling.

Keywords: Multibody dynamics, compliant actuators, contact dynamics, humanoid robot, Robotran, Simbody

1 INTRODUCTION

In this article, we present a multibody model of the COMAN humanoid robot† [1]. The key features of the proposed model are:

1. an efficient multibody dynamics allowing short simulation computational time;
2. the full electromechanical model of compliant actuators [2] made up with ordinary differential equations of the actuators inner dynamics;
3. a reliable mesh-to-mesh contact processing in order to simulate the robot self-collision and contacts with the environment.

For deriving the multibody equations, the Robotran symbolic generator was selected due to its reliability and efficiency [3]. This work further builds upon [4] by proposing a new actuator modeling and more powerful contact processing. Accurate mesh-to-mesh contacts were obtained through a coupling between the C-code provided by the Robotran generator and the C++ functions provided by the open-source library Simbody [5]. This model proved to be useful to speed up the synthesis and tuning of movement controllers, and can easily be adapted to the simulation of other robots.

†This work is supported by the European Community’s Seventh Framework Programme (FP7/2007-2013) under Grant 611832 (WALK-MAN).
The structure of the article is the following. First, we describe the multibody system (MBS) of the humanoid robot we have to simulate and control and describe our requirements for a simulator to accurately reproduce the behavior of this MBS. Then we compare three open-source multibody platforms used in robotics — that is the Robotran generator [3], Open Dynamics Engine ODE [6] and Simbody [5]. In the following section, we describe two approaches for contact simulation — i.e. rigid and compliant contact — and show our reasoning for choosing one of them. Based on this analysis, we derive the full MBS model of robot mechanics, which is augmented with its electromechanical model of the compliant actuators. Finally, we share the technical details for coupling the C-code of the Robotran generator with the contact module of the open-source physics engine Simbody.

2 STRUCTURE OF THE ROBOT AND SIMULATOR OVERVIEW

![MBS COMAN model](image)

The multibody model of COMAN consists of 24 absolutely rigid bodies and has a tree-structure shown in Figure 1. The root body Waist is a floating base and has 6 degrees of freedom (DoFs). The other 23 bodies (4 segments for the torso, 4 for each arm and 6 for each leg) are serially attached to their parents by revolute joints. So, the model has 29 mechanical DoFs (among which 6 DoFs for the floating base).

Similarly to the real robot, the model is equipped with 23 series elastic actuators [7] in the revolute joints, each producing a motor torque \( \tau_i, \ i = 1 \ldots 23 \). This torque depends on the corresponding joint angle, joint velocity and the motor angle \( q_m^i \) and velocity \( \dot{q}_m^i, \ i = 1 \ldots 23 \). Dynamics of this inner actuator variable \( q_m \) is governed by a differential equation involving the current \( I_i \) in the motor, which in turn satisfies a DC motor differential equation with a controllable voltage input.
Furthermore, the robot legs are equipped with tunable springs going in parallel to the motors [2]. These springs thus deliver the torques $\tau^p(p,q)$ at the pitch leg joints, where $p = (p_1, \ldots, p_6)$ is the controllable spring pretensions. Physical model of both types of actuators and differential equations that govern the variables $q^m = (q^m_1, \ldots, q^m_{23})$ are discussed in details in Section 4.

To accurately simulate this MBS, we need a simulator which combines several modules being connected as represented in Figure 1, on the right. The simulator core is a direct dynamics module (block A) deriving the acceleration of the MBS links based on the MBS structure and forces (external and control) applied to MBS. The way to represent of the MBS structure influences the speed of simulation as well as its accuracy (see Section 3).

A critical component of an integrated robotic simulator is an algorithm for contact processing (block B in Figure 1). Contacts between two segments of a robot (i.e. self-collisions) and with the environment (ubiquitous in walking or grasping), produce external forces that strongly influence the system dynamics. The inputs of block B are the state variables in the general sense, i.e. also including variables capturing the environment state, and the outputs are the contact forces. We discuss this block in Section 5.

To model the actuators used in COMAN, a simulator should provide a tool that allows to integrate the differential equations for the actuators internal state variables together with the MBS mechanical equations (so-called “strong coupling” for multiphysics systems [3]). For the series elastic actuators equipping this robot, these equations are those governing the motor current dynamics and external position. The inputs of this block (C in in Figure 1) are the control input (in our case voltages and spring pretensions) and the state variables. The outputs are the actuator forces and the derivatives of the actuator state variables.

A unified interface between the robot controller and simulator is very helpful for facilitating the development of the controller robot. Ideally, the developed controller should not depend on the specific command and data format of the simulator (Figure 1, block E) (see, e.g., [8, 9]). This would permit to easily transfer the controller code to the real robot or to other simulators, providing connections to a unique interface.

3 DIRECT DYNAMICS MODULE

The direct dynamics module aims at establishing the equation of motion of the MBS using a formalism based on Newton-Euler, virtual work or Lagrange equations (direct dynamics module). Here we focus on the influence of relative vs. absolute and symbolical vs. numerical approaches for the direct dynamics module.

3.1 Relative vs. absolute coordinates

The dynamic equations algorithm computes the MBS velocities and accelerations when the initial positions, velocities and the time evolution of the external and internal forces are given. To this respect, two approaches compete in the literature, namely those based on relative and absolute coordinates.

Most physics engines (e.g., Robotran, MuJoCo [10], and Simbody) use a minimal set of relative coordinates to determine the state of a multibody system and to avoid introducing algebraic constraints when possible. It means that for tree-structured multibody systems – in which bodies are connected without explicit constraints between the generalized coordinates and velocities – the number of generalized coordinates introduced in the system of equations is equal to the number of degrees of freedom of the system. To capture the connection between two adjacent bodies, for example by means of a revolute joint, only one coordinate and one velocity are needed – namely an angle and its time derivative. The set of Newton-Euler equations governing the dynamics of such a system is typically built by a recursive algorithm [11].

Alternatively, in the physics engines that were initially built for video-games and now are widely
used in robotics (e.g. Open Dynamics Engine (ODE) [6], Bullet [12]), absolute coordinates are used to specify the positions of the rigid bodies. It means that for the connection between two adjacent bodies (e.g. by means of a revolute joint), such an engine requires to introduce 6 DoFs per body and 5 bilateral algebraic constraints. This paradigm likely emerged because of the necessity to compute a lot of random impacts, ubiquitous in video games. The impacts were treated as simultaneously added algebraic unilateral constraints. However, this approach may lead to unrealistic behavior of the systems, constraint violations and inaccuracies (about artifacts in video-game engines and their reasons, see chapter 1.2 in [13]).

We carried out a comparison of absolute and relative approaches on a simple example of a compound pendulum (a mass, a length $l$ of a segment between the fixed point and the center of mass, gravity acceleration $g$ normalized to 1, diagonal inertia matrix with principal moments equal to 0.5). These dynamic equations were simulated in the ODE physics engine (absolute coordinates) and Robotran (relative coordinates). Since it is impossible to separate equation generation and numeric integration in ODE, motion was simulated for 25 units of time (one second equals one unit of time multiplied by $\sqrt{\frac{g}{l}}$, where $g$ and $l$ are in meters/sec$^2$ and meters, respectively). For ODE we used a built-in numeric integrator and the default parameters for the variables governing stability and accuracy (CFM equals $1 \times 10^{-10}$ and ERP equals 0.2, for more details see ODE wiki [14]). We used equations generated by Robotran in Matlab. The selected numeric integrator was the Runge-Kutta method of 4-5 order with time-varying step. The absolute and relative tolerances were equal to $2.2 \times 10^{-14}$ (ode45). As ground truth solution, the differential equations of the pendulum were integrated with the Matlab ode45 function with the same parameters. Initial conditions were set to get a continuous rotational motion of the pendulum (pendulum is oriented downward and the angular velocity equals 1.8). The full mechanical energy fluctuations and the angular coordinate error over time for Robotran are around $4 \times 10^{-13}$, i.e. the absolute tolerance multiplied by the simulation time. The error of the ODE solution depends on the time-step. The smallest error was achieved for a time-step of about $1 \times 10^{-7}$; constraint violation is around $6 \times 10^{-13}$; full mechanical energy fluctuations are around $7 \times 10^{-8}$. The time-curve for angular coordinate error is shown in Figure 2. The error grows up with time and cannot be decreased with smaller time-steps. Furthermore, it is likely that such errors would propagate over serial joints and segments for systems having more DoFs than the simple one presented here. Finally, this tolerance was achieved at the expense of a greedy simulation time (for ODE it took more than 5 hours to simulate 25 time units on a 3.4 GHz PC, for Robotran in Matlab environment, 7.4 sec only).

Figure 2. Absolute error of the angular coordinate and full mechanical energy of the compound pendulum for ODE (blue line) and Robotran (green line).
3.2 Symbolic vs. numeric generation

Again, two schools compete regarding the computation of the direct dynamic model, namely symbolic and numerical. A symbolic generation for a given system is run only once. Its output is a symbolic code providing the time derivatives of the state variables in analytical form. Examples of multibody simulators using symbolic algorithms are Robotran and MapleSim [15]. This approach permits to apply automatic symbolic simplifications on the dynamic equations (for example, Robotran performs trigonometric and recursive simplifications) during the equation generation. In the case of sparse mass matrices, the Robotran generator eliminates the useless terms. In the case of recursive simplifications, the full equations that are not used in the subsequent steps are also eliminated (for details, see [3]).

The automatic procedure for numeric generation of derivatives is the same for all multi-body systems, so it is impossible to perform ‘system-specific’ simplifications, as offered by the symbolic approach. This automatic rebuilding of the derivatives during time-integration is potentially helpful if event-based changes appear in the system, for instance if the system includes unilateral constraints like contacts. We will focus on the connection between the contact algorithm and symbolic/numerical algorithms in the following section.

![Figure 3. CPU time for the computation of generalized accelerations versus number of DoFs for the COMAN multibody system in Simbody (blue line) and Robotran (green line) physics engines.](image)

As an illustration of the potential superiority of the symbolic vs. numerical approach in direct dynamics module when accuracy and speed are required, we carried out a comparison between two representative algorithms of each family. The tested setup was a tree-like multibody system representing a humanoid robot (full description is provided in Section 2). This MBS was created in two different simulators: Robotran, adopting the symbolic approach, and Simbody, adopting the numerical one (both in relative coordinates), with identical inertial and geometric parameters. For both engines, we provided the same discrete random inputs: 29 joint positions, 29 joint velocities, and 23 generalized torques in the rotational joints. We then compared the produced outputs, i.e. the generalized accelerations, and computation time. The produced accelerations were thus not integrated to get the next velocity, in order to compare the computational time of the direct dynamics module only, disregarding issues related to the selection of a specific time integrator. These simulations revealed that the average relative difference in the generated accelerations was $1.9 \times 10^{-11}$%. Both physics engines are thus eligible. The computation speed for full COMAN
is approximately 3.5 times faster for the Robotran physics engine as compared to the Simbody one (on the same computer). Figure 3 shows an evolution of the computational time required to compute the joint accelerations, as a function of the number of degrees of freedom in this system. Comparison was performed for 12 (waist and one leg), 18 (waist and two legs), 21 (full torso and two legs, without arms), and 29 degrees of freedom. It shows that for both algorithms, computational time grows up as a linear function of the number of DoFs. Moreover, the Robotran slope is about 2 times smaller than the Simbody one. Therefore, the relative superiority of the symbolic vs. numeric approach further grows up with the number of DoFs.

4 DYNAMIC EQUATIONS AND COMPLIANT ACTUATORS’ MODELS

Based on the comparison and arguments provided in Section 3, the Robotran symbolic generator was selected for symbolical generation of MBS dynamic equation for DirectDynamics module together with the compliant contact processing. So, the second-order mechanical multibody model of the robot is provided by Robotran in the symbolic form:

\[ M(q)\ddot{q} + C(q, \dot{q}) + G(q) + N^c(q, \dot{q}) = T_1(q)\tau(q, \dot{q}, \ddot{q}, \dot{q}^m) + T_2(q)\tau^p(p, q), \]

where \( q = (q_1, \ldots, q_{29}) \) is the vector of angular joint positions plus cartesian coordinates of the floating base, \( M(q) \) is the mass-inertia matrix, \( C(q, \dot{q}) \) represents Coriolis and centrifugal forces, \( G(q) \) represents gravitational forces and torques. Contact forces and torques \( n(q, \dot{q}) \) have \( 6N^c \) components, where \( N^c \) is the number of active contacts. For walking, for example, we have \( N^c \) equal to 1 or 2 depending on a gate phase. Control is provided by the 23 serial actuators’ torques \( \tau(q, \dot{q}, \ddot{q}, \dot{q}^m) \) and 6 parallel actuators’ torques \( \tau^p(p, q) \). Control torques and contact forces propagate through the MBS by \( 29 \times 23, 29 \times 6 \) and \( 29 \times 6N^c \) matrices \( T_1(q), T_2(q), \) and \( N^c(q) \) correspondingly (they are all symbolically generated and simplified by Robotran).

Dynamic models of realistic actuators are often missing in existing simulators, where the joint actuators are rather considered as sources of pure torque or position (depending on the mode of control). The motor inertia when reflected to the output of gearbox often has the same order of magnitude as the link inertia. On the other hand, the robotics community recently promoted the use of compliant actuators to enhance the robot safety and energy efficiency, mainly when contacts with the environment are ubiquitous. Typically, these solutions require to design flexible joint robotic systems, where the electric motors are connected in series with a compliant element to mainly provide better force regulation and also shock absorption against environmental impacts.

A diagram of such actuators as used in [2] is shown in Figure 4. Therefore the links in robots with flexible joints are indirectly driven by the torque provided by the spring deflections.

For each actuated joint we have an equation that governs motor inner variable \( q^m_i \):

\[ J_i \ddot{q}_i^m + D_m \dot{q}_i^m + \tau_i(q_i, \dot{q}_i, \ddot{q}_i^m, \dot{q}_i^m) = K_i \dot{l}_i^m, \quad i = 1, \ldots, 23 \]

where \( J \) is the motor inertia (including rotor and the moving parts such as gearbox), \( D_m \) is the motor’s mechanical damping, \( K_i \) is torque constant and \( \dot{l}^m \) is the motor current. The motor load torque is given by

\[ \tau_i(q_i, \dot{q}_i, \ddot{q}_i^m, \dot{q}_i^m) = K_s(q_i - \dot{q}_i^m) + D_s(\ddot{q}_i - \dot{q}_i^m), \]

where \( K_s \) and \( D_s \) are stiffness and damping of serial spring respectively.

The current dynamics for \( i \)-th motor is modelled as

\[ L_i \dot{l}_i^m + R_i \dot{l}_i^m + K_o \dot{q}_i^m = V_i \]

where \( L, R, K_o \) and \( V_i \) are motor inductance, resistance, back EMF constant and applied input voltage.
If a parallel spring is further added, its torque is given in (5), where \( r \) is the radius of the joint’s pulley, \( k_p \) and \( d_p \) are the spring stiffness and damping. The leg pitch joints with a parallel spring are indexed with \( i \) where \( i = 7, 10, 12, 13, 16, 18 \) according to Figure 1.

\[
\tau^p_j(p_j, q_i) = \begin{cases} 
  k_p(p_j - rq_i) + d_p(\dot{p}_j - r\dot{q}_i), & \text{if } (p_j - rq_i) < 0 \\
  0, & \text{if } (p_j - rq_i) \geq 0 
\end{cases}, \quad j = 1, \ldots, 6. \tag{5}
\]

---

**Figure 4.** The series and parallel branch of a compliant actuator of the COMAN.

The last term that we need to describe here is the contact term \( n(q, \dot{q}) \). A contact dynamics module does not exist in the Robotran and has to be implemented. In the next section we provide some details on the algorithms and share our experience of coupling the Robotran simulator with external contact library.

## 5 CONTACT PROCESSING

The physics of contact is very complicated by itself [16]. Developers of contact modules face different types of problems, from the challenges of establishing the constitutive laws of impacts to the mathematical problems of stability and convergence of numerical methods. Imprecisions of the contact model and parameters introduce the largest inaccuracies in simulation [17]. Contact algorithms are quite expensive regarding computational cost. Moreover, a given algorithm that a simulator uses for contact modeling influences the choice of the associated numeric integrators and the internal representations of the equations of motion. Choosing a contact algorithm is thus challenging. Here we provide some reasoning to help making this choice. It is important to make a distinction between compliant and rigid contacts. Precise definitions and extended comparisons between these two approaches for contact modeling can be found in [18]. Here below, the pros and cons of rigid and compliant contact modeling are overviewed with a global point of view, i.e. with the objective to emphasize connections between the different contact modules with the equation formalism as discussed above.

### 5.1 Rigid Contact

Briefly speaking, rigid contact means that a contact between two (or more) bodies is treated as an instantaneously imposed unilateral constraint without interpenetration of bodies. So the inputs of such algorithm are the positions and velocities of bodies before contact, the constitutive impact law (i.e. the rule to compute the post-impact velocities – in the simplest case, through a restitution coefficient), and some properties of the system, like inertial matrices and bilateral constraints. The outputs of the algorithm are the velocities of all bodies after impact and, the corresponding
reaction forces. These velocities should satisfy all constraints (unilateral and bilateral). There is thus no interpenetration of the bodies and the mechanical compression and decompression phases are not explicitly calculated. The contact reaction forces – making the contact constraints to be satisfied – depend on other (bilateral or unilateral) constraints that are imposed to the system. Different algorithms exist for rigid contact processing (see [18], [19]): LCP (linear complementarity problem), NLCP methods (non-linear complementarity problem) and others. In fact all of them implement different numeric algorithms for solving the same systems of equations (sometimes called complementary slackness mechanical systems, see [18]).

We identified two bottlenecks related to this approach. First, regarding the constitutive impact law, for complicated systems and multiple impacts, a simple restitution coefficient may vary depending on the impact conditions. Second, rigid algorithms treat unilateral constraints as imposed to points and not to surfaces. So it is necessary to specify the points where the contact constraints (i.e. no interpenetration) are expected to be fulfilled. While this is reasonably easy to do for primitives like spheres and boxes, this is much more challenging for complex and possibly non-convex surfaces covering robots. The localization of the contact points in this case is a difficult mathematical problem. Also, rigid algorithms could introduce inaccuracies when used with some passive compliances like rubber covers of some parts of robots. Nevertheless these algorithms are fast and solve impacts in a single time step, ideally preventing penetration of the bodies. All ‘absolute coordinate’ engines (ODE, Bullet) and MuJoCo implement rigid contact algorithms.

5.2 Compliant Contact

Compliant contact algorithms integrate contact modelling within the time-dependent multibody equations, through a compression and decompression phase. A repelling force is calculated via a visco-elastic model using the relative distance and velocity between pairs of points on the surfaces of the bodies in contact (elastic foundation model, EFM) [5, 18]. In general, for two bodies bounded by triangular meshes $M_1$ and $M_2$, a resulting contact force and torque is a sum:

$$f_{contact}(q^c, \dot{q}^c) = 0.5 \sum_1 (f_{normal} + f_{tangent}) + 0.5 \sum_2 (f_{normal} + f_{tangent})$$  \hspace{1cm} (6)

$$\tau_{contact}(q^c, \dot{q}^c) = 0.5 \sum_1 \textbf{r}_i \times (f_{normal} + f_{tangent}) + 0.5 \sum_2 \textbf{r}_i \times (f_{normal} + f_{tangent})$$  \hspace{1cm} (7)

The summation $\sum_1$ goes through the triangles of the mesh $M_1$ that overlap the mesh $M_2$ and vice versa for $\sum_2$. Forces $f_{normal}$ and $f_{tangent}$ are normal and tangent components of elementary repelling force applied to the current triangle. They depend on relative propagation and velocity of the corresponding triangle with respect to the opposite mesh; $\textbf{r}_i$ is a vector from a center of a contact patch to the center of the triangle. Thus, such an algorithm calculates a force and a torque (applied to the computed center of patch) from the positions and velocities of both bodies in contact, i.e. $q^c$ and $\dot{q}^c$. Models for $f_{normal}$ and $f_{tangent}$ can vary for particular models (visco-elastic models, viscous, dry, or Stribeck friction, etc).

The main shortcomings of this approach are (i) the difficulties of selecting the appropriate parameters for the elastic layer; and (ii) a ubiquitous trade-off between keeping the interpenetrations of the bodies within reasonable limits and maintaining the computational cost low enough. Indeed, the stiffness of the associated differential equations is directly correlated with the interpenetration magnitude: for high stiffness $k$, the bodies interpenetrations are of order $O(k^{-2/3})$ and the time-step is of order $O(k^{-1/2})$ [18]. So, the computational speed and accuracy critically depends on this parameter. The EFM algorithm is faster and at the same time well approximates finite elements algorithms (FEM), which are considered as an etalon for the deformable bodies [20], [21]. EFM algorithms can simulate interpenetration of the body pairs, micro-slips, and repeated impacts which also occur in the real world.

To summarize, rigid contacts are fast and generate zero or negligible interpenetration of the bodies. This approach is more suitable for perfectly rigid bodies, for example for stainless balls, or railway
wheels. It can be adapted to real-time applications when some lack of accuracy can be permitted. Rigid contact modeling can also produce incorrect output, mainly for complex body shapes. Since it is difficult to separate the direct dynamics module from the contact one, the potential inaccuracy source is usually challenging to identify. This approach can further produce inaccuracies in simulations of robots with passive compliance, for example, covered by a deformable layer. Compliant contact algorithms are more expensive regarding computational load and imply additional stiffness to the numerical solutions of the ordinary differential equations. These algorithms can be easily isolated from the other parts of the simulator and thus coupled to any core of the multibody simulator. They are more accurate for deformable bodies such as rubber feet and hands, if appropriate parameters are well estimated. These algorithms require more computational power to be executed in real-time applications, although not out of the capacities of modern computers.

We propose to use for contacts forces \( \mathbf{n} \) in Equations (1) the compliant model for contacts (6,7):

\[
\mathbf{n} = (f_1^{\text{contact}}, \tau_1^{\text{contact}}, \ldots, f_N^{\text{contact}}, \tau_N^{\text{contact}})
\]

Here, we augment Robotran simulator with external open-source library providing compliant contact. This offers a general algorithm for handling contact in the Robotran framework, especially for complex objects whose shape requires to be modeled by a polygonal mesh. In the next section we explain how we coupled the Simbody compliant library with the MBS equations of motion generated by Robotran.

### 5.3 COUPLING ROBOTRAN SIMULATOR WITH COMPLIANT CONTACT LIBRARY

![Diagram of coupling between Robotran and Simbody](image)

**Figure 5.** Scheme of coupling between Robotran and Simbody.

Contact processing was made possible by coupling Robotran with an external C++ contact library
developed by the Simbody team [5]. This library was selected based on the following reasons:

- It is open-source and is distributed under permissive Apache 2.0 License;
- It provides compliant contact of two types: Hertz-model and elastic foundation model (EFM, more details can be found in [5, 21]).
- It allows to use WaveFront object files encoding triangular meshes to define complex body shapes;

The scheme of coupling is shown in Figure 5. Concretely, during the building of the multibody structure (that has to be constructed in the Robotran’s GUI-editor MBSPad), the user needs to add S-sensors and F-sensors for each body which will have contact surfaces (i.e. the so-called contact bodies, CBs). A “S-sensor” is a Robotran tool that is used for mapping relative coordinantes and velocities to absolute coordinates and velocities of the CBs. A “F-sensor” is a Robotran tool that allows to define user-external forces. The Robotran server then generates both the MBS equations of motion, and the coordinate transforms being required to manage contacts through S-sensors and F-sensors. The user further needs to specify the number of CBs, the physical properties governing contact (such as stiffness, viscosity, friction coefficients) for each of them (including ground) and to define the surface of contact, through primitives or meshes, in a C-code template.

The execution starts with the initialization of all the variables that are used during the simulation loop. An instance of the Simbody class world is created. In this world a shadow free body with 6 DoFs for each CB is created. The computation is thus separated between Robotran (in charge of the dynamic integration, complying with the mechanical constraints between the successive bodies) and Simbody (in charge of the computation of the contact forces through the 6-DoFs shadow bodies). Simulating COMAN walking on a rough terrain requires creating only two shadow bodies, i.e. one for both feet, with mesh contact surfaces and a ground in the Simbody world.

In the simulation loop, relative positions and velocities are transformed into absolute positions and velocities, through the code generated for the S-sensors. These positions and velocities are then imposed for each shadow CB in the Simbody world. Then the Simbody library processes the contact and returns values of external forces and moments due to contacts. These forces and moments are then applied to the MBS in Robotran (relying on the F-sensors) and included to update the bodies dynamics.

In Figure 6 two representative tasks that extensively use 3D mesh-to-mesh contact are shown: that are a manipulation task and a locomotion task (with a gait controller presented in [22]).

![Figure 6](image-url)

**Figure 6.** On the left: Manipulation task – COMAN takes a box from a table. On the right: Locomotion task – COMAN goes over a 3D bump.
6 CONCLUSIONS

In the frame of this work, a fast and accurate model of a humanoid robot is derived. Comparison of time-efficiency of different method for direct dynamics module is provided. The full electromechanical models for series compliant actuators and for parallel elastic branch are derived. Coupling of symbolic MBS differential equations with open-source external compliant contact library is made. The simulator that integrates over time the presented multiphysics model shows reliable results for locomotion and manipulation tasks.

REFERENCES


On Attitude Dynamics and Control of Legged Robots Using Tail-Like Systems

Konstantinos Machairas and Evangelos Papadopoulos

School of Mechanical Engineering
National Technical University of Athens
9 Heroon Polytechniou Str., 15780 Athens, Greece
[kmach, egpapado]@central.ntua.gr

ABSTRACT
In this work we study the attitude dynamics and the control of legged robots using tail-like appendages during the aerial phases of high speed locomotion. A free floating two-body system is used to describe the dynamics of a large body controlling its attitude using a rotating appendage. The equations of motion for a tail and a reaction wheel are given, and the meaning of the generalized coordinates being ignorable or palpable is discussed in detail. A thorough discussion on the holonomy of the system is also included. Analytical expressions are given for a further reduced dynamical model and model-based controllers are then proposed. Finally, we present a series of simulation results, and we derive conclusions that can serve as guidelines when designing such systems.

Keywords: Legged Robot, Tail, Reaction Wheel, Attitude Control, Nonholonomic.

1 INTRODUCTION
Over the last five years, research in legged robotics has drawn again the attention of the robotics research community, as universities and companies came up with impressive accomplishments, especially in the field of quadrupedal locomotion. Although much research has been conducted concerning the design and control of legs of various morphologies, attitude control of the body is yet poorly investigated. However, most of the tasks assigned to legged robots, such as high speed galloping or jumping over obstacles, require precise control of the robot’s attitude. So far, attitude control is mostly achieved indirectly through the motion of the legs, a technique that assigns more tasks to the legs forcing them to trade-offs that may lead to low performance.

To better mitigate this challenge, dedicated appendages with greater moment of inertia (MoI) can be used. Interesting ideas can be derived from biology; one quickly thinks of animal tails. Many quadruped mammals have long tails, which aid to balance and maneuver at high speeds, [1]. Kangaroo rats use their long tails for righting and turning in midair. Black rats can impressively enter a building by balancing along a 2mm wire. Moreover, studying hopping by kangaroos, one may be amazed to see how they use their tails to counteract the body pitching induced by the motion of their legs, [2]. In general, legged animals mostly use their tails for fine adjustments to perturbations, when their legs are otherwise occupied.

While numerous legged robots have been designed, only a minority employ dedicated appendages for angular momentum management, such as tails or reaction wheels, [3-7]. A number of studies have also dealt with attitude control under conservation of angular momentum, and methods that can lead a mechanism from an initial configuration to a desired final one have been developed, [8-10]. However, to the authors’ knowledge, no methodology has been proposed concerning the design of such tail-like mechanisms.

In this work, we use a free floating two-body system to describe the dynamics of a large body controlling its attitude using a rotating appendage. The equations of motion are next given for all cases, and the meaning of the ignorable and palpable coordinates is discussed in detail. To this end, we also clarify several issues concerning the holonomy of the system and its implications on attitude control. The model is then further reduced and model-based controllers...
are proposed. Finally, we present a series of simulation results, and we derive conclusions that can serve as guidelines when designing such systems.

2 DYNAMICS AND ANGULAR MOMENTUM

2.1 Dynamics

We introduce a simple planar template of two coupled bodies, i.e. a body and a tail in aerial phase (see Table 1). By body we mean the body with the four legs and the head of a legged robot, except for the tail. This is a reasonable assumption if one considers zero leg MoI and a rigid spine. The uniform gravitational field allows separation of the system center of mass (CoM) motion and the relative motion into two decoupled pieces. Therefore we can decouple the system CoM motion, and obtain a reduction to the system CoM frame. We parameterize the configuration space only by the absolute pitch angle of the body \( \theta \in S^1 \), and the relative hinge angle of the tail \( q \in S^1 \). Let \( (m_s, I_s) \) and \( (m_l, I_l) \) denote the mass and the MoI about each body CoM, for the body and the tail respectively. Let \( r \) be the distance from the body CoM to the joint, and \( l \) be the distance from the tail CoM to the joint. Finally, let \( \tau \) be the control torque that the body exerts on the tail, with the motor modelled as an ideal torque source.

The equations of motion (EoM) are given in Table 1 for every possible case, including both reaction wheel and tail cases, with a reaction wheel being any symmetrical body hinged at its own CoM. We note that the two masses appear only in the form of an important quantity \( \mu = (m_s m_l) / (m_s + m_l) \), that we call the system effective mass. In all cases, the generalized coordinates are characterized as ignorable or palpable, since this distinction can help the analysis. A coordinate is called ignorable or cyclic when it does not appear in the Lagrangian, and palpable or positional otherwise. In practice, when a coordinate is ignorable, we can write the EoM without this coordinate. We note that when the hinge is transferred to the body CoM, the shape angle \( q \) turns from palpable to ignorable, with important implications on the system’s holonomy, which are discussed next thoroughly.

2.2 Conservation of Angular Momentum

We note that the generalized momentum associated with the ignorable coordinate \( \theta \) is conserved \( \left( \partial L / \partial \dot{\theta} = \text{const} \right) \), where \( L \) is the Lagrangian, yielding:

\[
(I_s + \mu r^2 + I_l + \mu l^2 + 2\mu rl \cos q)\dot{\theta} + (I_s + \mu l^2 + \mu rl \cos q)\dot{q} = h_0
\]

which is in fact the equation for the conservation of the system’s angular momentum about its CoM, with \( h_0 \) being the system initial angular momentum.

2.3 Integrability of the Constraint and System’s Holonomy

As thoroughly discussed in [11], the general case of a planar free-floating open kinematic chain is nonholonomic for \( n > 2 \) bodies. However, in literature, the two-body system is often incorrectly considered either holonomic or nonholonomic without the appropriate analysis. In this work, the problem is addressed in detail, and it is shown that the system’s holonomy depends on the system's geometry and the system's initial angular momentum.

Equation (1) can take the form of an acatastatic Pfaffian constraint, which is nonholonomic only when \( r, l \neq 0 \) and \( h_0 \neq 0 \) at the same time. This means that for zero initial angular momentum, the conservation equation is analytically integrable independent of the position of the hinge. When time enters as a third variable through the initial angular momentum, the constraint is integrable only if the tail is pinned at the body CoM \( (r = 0) \), or if the appendage is a reaction wheel. A holonomic constraint is in fact a geometric one and thus each \( \theta \) corresponds to a specific \( q \), while a nonholonomic constraint makes the whole configuration manifold accessible,


<table>
<thead>
<tr>
<th>General case: 2-body free floating system</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\theta$: ignorable, $q$: palpable</td>
</tr>
<tr>
<td>$(I_0 + \mu r^2 + I_r + \mu l^2 + 2 \mu r l \cos q) \ddot{\theta} + (I_r + \mu l^2 + \mu r l \cos q) \dot{q}$</td>
</tr>
<tr>
<td>$- \mu r l \sin q (\dot{q}^2 + 2 \dot{q} \ddot{q}) = 0$</td>
</tr>
<tr>
<td>$(I_r + \mu l^2 + \mu r l \cos q) \ddot{\theta} + (I_r + \mu l^2) \dot{q} + \mu r l \sin q \dot{q}^2 = \tau$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Tail hinged at distance $r$ from body CoM ($I_1 = 0$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\theta$: ignorable, $q$: palpable</td>
</tr>
<tr>
<td>$(I_0 + \mu r^2 + \mu l^2 + 2 \mu r l \cos q) \ddot{\theta} + (\mu l^2 + \mu r l \cos q) \ddot{q}$</td>
</tr>
<tr>
<td>$- \mu r l \sin q (\dot{q}^2 + 2 \dot{q} \ddot{q}) = 0$</td>
</tr>
<tr>
<td>$(\mu l^2 + \mu r l \cos q) \ddot{\theta} + \mu l^2 \dot{q} \ddot{q} + \mu r l \sin q \dot{q}^2 = \tau$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Tail hinged at body CoM ($I_1 = 0$, $r = 0$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\theta$: ignorable, $q$: ignorable</td>
</tr>
<tr>
<td>$l \dot{\theta} = -\tau$</td>
</tr>
<tr>
<td>$\frac{l_1 \mu l^2}{I_0 + \mu l^2} \ddot{q} = \tau$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Reaction wheel hinged at distance $r$ from body CoM ($l = 0$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\theta$: ignorable, $q$: ignorable</td>
</tr>
<tr>
<td>$(I_0 + \mu r^2) \ddot{\theta} = -\tau$</td>
</tr>
<tr>
<td>$\frac{l_1 (I_0 + \mu r^2)}{I_0 + I_r + \mu r^2} \ddot{q} = \tau$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Reaction wheel hinged at body CoM - “Elroy’s Beanie”, [12], ($l = r = 0$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\theta$: ignorable, $q$: ignorable</td>
</tr>
<tr>
<td>$l \dot{\theta} = -\tau$</td>
</tr>
<tr>
<td>$\frac{l_1 l_0}{I_0 + I_r} \ddot{q} = \tau$</td>
</tr>
</tbody>
</table>

and any pair $(\theta, q)$ can be achieved. Next, we present the holonomic cases, and we give analytical results based on the analysis in [13].

**Zero Initial Angular Momentum**

In this case, the conservation equation is integrable for all geometries. Integrating (1) with $h_0 = 0$ yields:

$$\theta = \theta_0 - \frac{1}{2} (q - q_0) + \frac{A}{C} \tan^{-1} \left( \frac{B}{C} \frac{q}{2} \right) + \frac{A}{C} \tan^{-1} \left( \frac{B}{C} \frac{q_0}{2} \right)$$
where
\[ A = I_0 + \mu^2 \, I_0 - 2 \mu r^2, \quad B = I_0 + I_1 + \mu (l-r)^2 \]
\[ C = \sqrt{(I_0 + I_1 + \mu^2 r^2)^2 - (2\mu l)^2} \]
This is a rather involved expression that gets much simpler when \( r = 0 \) (the appendage is hinged at the body’s CoM), or \( l = 0 \) (the appendage rotates about its CoM, i.e. it is a reaction wheel).

**Nonzero Initial Angular Momentum**

For nonzero angular momentum, the more general case in which (1) is integrable is when the appendage is hinged at body CoM. Integrating (1) with \( r = 0 \) yields:
\[ (I_0 + I_1 + \mu l^2)(\theta - \theta_0) + (I_1 + \mu l^2)(q - q_0) = h_0(t - t_0) \] (8)

**Conclusions**

- When both generalized coordinates are ignorable, the conservation equation is always integrable and the system holonomic.
- When both coordinates are ignorable, the inertia matrix becomes independent of the shape variable \( q \), the EoM can be written decoupled, and analytical solutions can be derived.
- When the initial angular momentum is zero, the system is holonomic for every geometry. Hence, it is not possible to achieve any pair of \( \theta \) and \( q \).
- When the appendage is a reaction wheel, the system is always holonomic.
- The system is nonholonomic only when a tail \( (l \neq 0) \) is hinged at a distance \( r \neq 0 \) from the body CoM and the initial angular momentum is nonzero.

3 REDUCED DYNAMICS AND CONTROL

Being difficult to control both \( \theta \) and \( q \) with a single control input \( \tau \), we develop model-based controllers to control \( \theta \) when we need to control the body attitude, and \( q \) when we need to position the tail to a desired angle.

### 3.1 Control of the Tail Angle \( q \)

For all cases \( \theta \) is an ignorable coordinate; we can derive the reduced EoM in the form of a single equation where only \( q \), \( \dot{q} \), and \( \ddot{q} \) appear (see Appendix for the full expressions):
\[ D(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q, h_0) = \tau \] (9)

Using (9) the following feedback linearization control scheme can be applied in order to control the tail angle \( q \), where \( e_q \) is the error in tail angle, and \( k_v, k_p \) are the gains of a PD controller:
\[ \tau = D(q)\ddot{q}_d + k_v e_v + k_p e_q \] (10)

### 3.2 Control of the Unactuated Body Angle \( \theta \)

In order to control \( \theta \) one should eliminate \( \ddot{q} \) from the second EoM (1), yielding a single equation of the form (see Appendix for the full expressions):
\[ D^*(q)\ddot{\theta} + C^*(q, \dot{\theta}) = \tau \] (11)

Similarly to the previous case, a model-based controller is developed for \( \theta \), achieving \( \ddot{\theta} = \ddot{\theta}_{\text{des}} \):
\[ \tau = D^*(q)(\ddot{\theta}_d + k_v \dot{e}_v + k_p e_v) + C^*(q, \dot{\theta}) \] (12)
Trajectory planning is implemented using a quintic polynomial of the following form (see Appendix for the full expressions):

\[ \theta_{ref}(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5 \]  

(13)

The control scheme described here was used in all the experiments included in this work.

4 DESIGN PRINCIPLES

The analysis above is important for the attitude control of a legged robot, and provides the basic guidelines for building a design methodology for tail-like systems. On this basis, steps for selection of the key parameters of reaction wheels and tails have been proposed in [13]. In this section we focus on tail design, since a tail has been proved to be more effective than a wheel, [13]. The main parameters to be selected are the mass and the length of the tail, while the motor characteristics are also of great importance. Depending on the case, several criteria can be used for the calculation of these parameters. These concern: (a) the maximum change of the body angle that can be achieved during a flight phase, (b) the maximum body angular velocity that can be rejected through the tail’s motion, (c) the maximum accelerating or decelerating force appearing at the tail joint, and (d) the maximum change on body’s angular momentum induced by leg motion. The design principles introduced herein are mainly based on the first criterion.

Suppose the body needs to perform a maneuver \( \Delta \theta \) in a specific time interval \( \Delta t \) in roll, pitch or yaw direction. In this case, a suitable torque profile \( \tau(t) \) must be provided by the tail-like appendage to rotate the body. The device can only deliver the torque needed, if a motor can accelerate the appendage by exerting on it the opposite torque. As expected, when the desired torque is exerted on a low MoI appendage, high angular acceleration and thus high angular velocity results, and this greatly affects the characteristics that the motor should have. For instance, a demanding task requires the motor to work in high torque (in order to rotate the body), and high speed (the speed that the lower MoI appendage reaches). Besides, the appendage's mass must be the lowest possible, so that a robot can use it without significantly increasing its total mass. These facts reveal the importance of the proper appendage design, and how this affects the selection of the driving motor.

In most cases, the rotation of the tail is mechanically constrained, as also observed in animal tails, and thus every maneuver must be completed while the appendage is within its mechanical bounds, i.e. \( q \in [q_{\text{min}}, q_{\text{max}}] \). Considering the conservation equation in the zero initial angular momentum case, when \( \Delta q \) is bounded, \( \Delta \theta \) is also bounded, i.e. not every desired maneuver can be performed. Hence, the parameters consisting this equation are significant for a proper design and determine the capabilities of the final device. On this basis, an expression for the tail mass calculation has been proposed in [13], which is also used in this work. At this point, we reach the following conclusions that we will validate through simulation experiments in section 5:

- By choosing the tail mass after maximizing the tail length, one chooses the maximum body maneuver \( \Delta \theta \) that can be performed when the system total angular momentum is zero, or the maximum initial body angular velocity that can be rejected in a time interval \( \Delta t \), when the system initial angular momentum is nonzero.

- The time interval to complete a maneuver is determined by the torque provided by the tail; the higher the torque, the faster the maneuver. Moreover, the lower the tail mass, the higher the tail acceleration under a certain torque profile, and therefore, the greater the power needed.

5 SIMULATION EXPERIMENTS

In our previous work, we have shown that a tail hinged at distance \( r \) from the body CoM is a better solution than a reaction wheel hinged at the same position; since the required motor power, and torque are significantly reduced in the tail case, [13]. Therefore, in this section, we present a series of simulation experiments of bodies performing maneuvers with tails of various
morphologies, to better understand the nature of the problem. Zero initial angular momentum was considered in all experiments, in order to reach conclusions easier. We were mostly interested in the control torque profile, and the maximum tail angular speed since, these are the parameters that mainly determine the motor selection. Considering DC actuators, the selection of a motor-gearbox-amplifier combination that can perform such maneuvers, is difficult or sometimes impossible. The speed – torque characteristics for a demanding maneuver often exceed the capabilities of typical DC actuators, and even if there is a suitable actuator for the task, its mass can be unacceptable. These difficulties lie mainly on the need to control the attitude of a large MoI body by rotating a small MoI appendage in a very short time interval. These facts make the following analysis valuable, since the limits regarding the actuators and the possible maneuvers are revealed through numerous experiments. The parameters of the simulations were chosen according to data obtained from animal and robot locomotion, [13].

5.1 Experiments varying the Tail Mass

First, we consider a body of mass $m_0=30$kg and MoI $I_0=2$kgm$^2$, performing a $\Delta \theta=3^\circ$ maneuver in 0.15s using a tail hinged at distance $r=0.4$m from the body CoM, with tail length $l=0.4$m, tail mass varying from 0.5 to 4 kg, and $I_1=0$. We use the expression given in [13] to calculate a minimum value for the tail mass, and with this in mind, we try greater values to see how other parameters, such as motor torque, speed, and power change. This is the first experiment, since the mass of the tail is the easiest parameter to change in a real robot. Simulation results are shown in Fig. 1.

Conclusions

In Figures 1(c) and 1(d), we see that for greater tail mass the maximum motor speed decreases, while the maximum torque slightly increases. Furthermore, the power that the motor has to deliver is much greater for lower tail mass, see Fig. 1(f). We conclude that the greater the tail mass, the better for the actuator, provided that the maximum torque can be supplied by the actuator, and the extra mass can be carried by the robot. Therefore, a good choice for the tail mass would be $m_1=1.5$kg, and thus this is the value used in the following experiments.

5.2 Experiments varying the Tail Length

Except for changing the tail mass, another way to change the tail MoI about its hinge is by changing the tail length $l$. In this series of simulations, we use the same parameters as above, with a tail mass of 1.5kg, and a tail length varying from 0.2m to 0.5m, to achieve a $\Delta \theta=3^\circ$ maneuver in 0.15s. Simulation results are shown in Fig. 2.

Conclusions

We reach similar conclusions with the previous case, in which we varied the tail mass, i.e. the greater the tail mass, the better for the actuator, provided that the maximum torque can be supplied, see Fig. 2(e). However, in this case, a limit exists for the tail length mostly due to the robot's geometry. For instance, it cannot be much greater than the leg's length.

5.3 Experiments varying the Body CoM – Hinge Distance

In this section we present simulation results from experiments of a $\Delta \theta=10^\circ$ maneuver in $\Delta t=0.2$s, and different hinge positions, varying the body CoM – hinge distance $r$ from 0m to 0.45m. The rest of the parameters are kept similar to the ones in the previous simulations. The results are shown in Fig. 3.
Figure 1. A body of \( m_0=30 \text{kg}, I_0=2 \text{kgm}^2 \), with a tail of length \( l=0.4 \text{m} \), MoI \( I_1=0 \), and mass \( m_1 \) varying from 0.5kg to 4kg, hinged at distance \( r=0.4 \text{m} \) from the body CoM, performs a maneuver of \( \Delta \theta=3^\circ \) in \( \Delta t=0.15 \text{s} \).

Conclusions

As shown in Figures 3(c), 3(d), and 3(f), the torque, speed and power profiles are symmetric when the tail is hinged at the body CoM. In every other case, where \( r \neq 0 \), the hinge force creates a torque that breaks this symmetry and helps the motor perform the maneuver. As a result, the motor torque, speed and power decrease as the body CoM – hinge distance increases, i.e. the greater this distance is, the easier for the actuator to perform a certain maneuver.

5.4 Experiments varying the Time of the Maneuver

It is evident so far that the time interval chosen for a certain maneuver is a key parameter of the task, and strongly affects the motor selection. In this series of simulations we address this topic systematically by performing a \( \Delta \theta=10^\circ \) maneuver in different time intervals \( \Delta t \), varying from 0.15s to 0.4s. The body and tail parameters are: \( m_0=30 \text{kg}, m_1=1.5 \text{kg}, I_0=2 \text{kgm}^2, I_1=0, l=0.4 \text{m} \), and \( r=0.4 \text{m} \). The results are presented in Fig. 4.
Figure 2. A body of $m_0=30\text{kg}$, $I_0=2\text{kgm}^2$, with a tail of mass $m_1=1.5\text{kg}$, $I_1=0$, and length $l$ varying from 0.2m to 0.5m, hinged at distance $r=0.4m$ from the body CoM, performs a maneuver of $\Delta \theta=3^\circ$ in $\Delta t=0.15\text{s}$.

Conclusions

Figures 4(c), 4(d), and 4(f) show that by decreasing the time available for the maneuver, the motor power, torque and speed increase. The very short time interval is the main reason for the need of powerful motors that would be of no use for longer time intervals. The torque and power requirements decrease significantly even for a desired time of 0.4s. Moreover, the tail angle – time diagram reveals the time invariance of the holonomic angular momentum constraint (note that the initial angular momentum is zero in this experiment), since the change in tail angle is the same for all experiments, see Fig. 4(b).

5.5 Experiments varying the Body MoI

Legged robots are systems that must be able to move successfully with varying inertia properties. Two typical reasons are the need to carry different cargos, and the uncertainty regarding the knowledge of the robot's real inertia properties. This fact justifies the analysis presented in the section, which includes simulation experiments with bodies of different MoI controlling their attitude with identical tails. Maneuvers of $\Delta \theta=3^\circ$ in $\Delta t=0.15\text{s}$ are performed by
Figure 3. A body of $m_0=30$kg, $I_0=2$kgm$^2$, with a tail of mass $m_1=1.5$kg, MoI $I_1=0$, and length $l=0.4$m, hinged at a distance $r$ from body CoM, which varies from 0m to 0.45m, performs a maneuver $\Delta \theta=10^\circ$ in $\Delta t=0.2$s.

bodies of MoI from 1.6 to 2.4 kgm$^2$. The other simulation parameters are: $m_0=30$kg, $m_1=1.5$kg, $I_1=0$, $l=0.4$m, and $r=0.4$m. The results are shown in Fig. 5.

Conclusions
As expected, for bodies of greater MoI, greater power, torque and speed are requested from the motor, see Figures 5(c), 5(d), and 5(f). Hence, the greater the body MoI, the harder for the actuator to perform a certain maneuver, see Fig. 5(e).
Figure 4. A body of $m_0=30\text{kg}$, $I_0=2\text{kgm}^2$, with a tail of mass $m_1=1.5\text{kg}$, $M_1=0$, and length $l=0.4\text{m}$, hinged at a distance $r=0.4\text{m}$ from the body CoM, performs a $\Delta\theta=10^\circ$ maneuver in time intervals $\Delta t$ varying from 0.15 to 0.4s.

6 CONCLUSIONS

In this paper we studied the attitude dynamics and the control of legged robots using tail-like appendages during the aerial phases of high speed locomotion. A free floating two-body system was introduced to describe the dynamics of a large body controlling its attitude using a rotating appendage. The equations of motion for a tail and a reaction wheel were given, and the meaning of the ignorable and palpable coordinates of the system was discussed in detail. The holonomy of the system was also discussed thoroughly. Analytical expressions were given for a further reduced dynamical model and model-based controllers were proposed. A series of simulation experiments were finally carried out for various system parameters, and important conclusions were derived concerning the design of such systems.
Figure 5. A body of $m_0=30\text{kg}$, and $\text{MoI}_0$ varying from 1.5 to 3.5kgm$^2$, with a tail of length $l=0.4\text{m}$, $\text{MoI}_1=0$, and mass $m_1=1.5\text{kg}$ hinged at $r=0.4\text{m}$ from body CoM performs a $\Delta \theta=3^\circ$ maneuver in $\Delta t=0.15\text{s}$.

ACKNOWLEDGMENT

This research has been financed by the European Union (European Social Fund – ESF) and Greek national funds through the Operational Program “Education and Lifelong Learning” of the National Strategic Reference Framework (NSRF) – Research Funding Program: ARISTEIA: Reinforcement of the interdisciplinary and/ or inter-institutional research and innovation.

REFERENCES


APPENDIX

Terms of the reduced EoM (9), when written as functions of \( q, \dot{q}, \ddot{q} \):

\[
D(q)=\frac{(d_{00}d_{11}-d_{01}^2)}{(d_{10}+d_{11}+2d_{10})},
\]

\[
C(q,\dot{q})=\frac{(d_{00}+d_{10})(d_{11}+d_{10})\ddot{q}}{(d_{00}+d_{11}+2d_{10})^2},
\]

\[
G(q,\dot{q},\ddot{q})=\ddot{q}^2/(d_{00}+d_{11}+2d_{10})^2,
\]

\[
d_{00}=I_0+\mu \dot{r}^2, \quad d_{11}=I_1+\mu \dot{l}^2, \quad d_{10}=\mu r \cos \theta, \quad \ddot{d}_{10}=\mu r \sin \theta
\]

Terms of the reduced EoM (11), when written as functions of \( q, \dot{\theta}, \ddot{\theta} \):

\[
D^*(q)=-\frac{d_{10}(d_{00}+d_{11}+2d_{10})}{d_{11}+d_{10}},
\]

\[
C^*(q,\dot{q},\dot{\theta})=\frac{d_{10}\ddot{q}^2}{d_{11}+d_{10}},
\]

With \( d_{00}, d_{11}, d_{10} \) and \( \ddot{d}_{10} \) defined above

Terms of the quintic polynomial used for trajectory planning in (13):

\[
a_0=\dot{\theta}_0, \quad a_1=\ddot{\theta}_0, \quad a_2=\frac{1}{2}\theta_0
\]

\[
a_3=\frac{1}{2r_f}(20(\dot{\theta}_f-\dot{\theta}_0)-(8\dot{\theta}_f+12\dot{\theta}_0)\alpha_f-3(\theta_0-\dot{\theta}_0)\dot{\theta}_f^2)
\]

\[
a_4=\frac{1}{2r_f}(-30(\dot{\theta}_f-\dot{\theta}_0)+(14\dot{\theta}_f+16\dot{\theta}_0)\alpha_f+(3\theta_0-2\dot{\theta}_0)\dot{\theta}_f^2)
\]

\[
a_5=\frac{1}{2r_f}(12(\dot{\theta}_f-\dot{\theta}_0)-6(\dot{\theta}_f+\dot{\theta}_0)\alpha_f+(\dot{\theta}_0-\dot{\theta}_0)\dot{\theta}_f^2)
\]
On the Use of the Wrench Exertion Capability as a Performance Index for Cable Driven Robot

G. Boschetti, A. Trevisani

Dept. of Management and Engineering (DTG)
University of Padova
Stradella San Nicola 3, 36100 Vicenza, Italy
[giovanni.boschetti, alberto.trevisani]@unipd.it

ABSTRACT

The evaluation of the performances of cable driven robots must explicitly take into account the considerable limitations introduced by cable tension bilateral bounds, basically translating the physical need that neither negative nor limitless cable forces can be exerted. Therefore, dedicated performance indices have to be developed for such robots. This paper introduces a novel performance index called Wrench Exertion Capability (WEC) which can be applied to any cable robot topology and provides an evaluation of the performances of a robot along a direction of interest in terms of force or torque exertion capabilities. The WEC is computed by solving a linear programming problem involving cable tensions, cable tension limits, and a novel representation of the so-called wrench matrix. The WEC index can also be employed to evaluate isotropy and the minimum performance that can be assured by a cable robot throughout the workspace, irrespective of the direction. Hence it represents a useful tool for proper robot design, comparative analysis, or robot positioning within a workcell. The computation of the WEC is here discussed with reference to both fully actuated or redundant cable robots and underactuated ones. Practical examples of WEC computation are provided to demonstrate its usefulness in the analysis and comparison of alternative cable robot topologies.

Keywords: Cable Driven Robot, Performance Index, Wrench Exertion Capability.

1 INTRODUCTION

Cable driven robots are parallel robots, operating in planar or spatial arrangements, obtained by attaching multiple cables to a moving platform, on which the end-effector is fitted. The cables are usually active in the sense that they are driven by motors which can extend or retract the cables by winding or unwinding them from pulleys (also called winches or drums). Cable driven robots, henceforth named cable robots for brevity, have been studied thoroughly since the early 90’s [1] and promise to significantly increase performances of today’s industrial robots in terms of payload, workspace and dynamic performances: they can be designed to have a very large workspace, a very high load capacity, or to generate very high speed motions [2], always with considerable energy efficiency. Their unique features, arising from parallel kinematics combined to minimal moving masses, make them amongst the most promising robotic devices in the industrial and service field, as it is proved by the ever growing number of cable robot families that has been developed by research institutions and private companies [3].

Very often, cable robots are designed to be redundant (i.e. with more active cables, and hence motors, than degrees of freedom, dofs, of the moving platform, see e.g. [2]), however, fully actuated (i.e. with a number of active cables equal to the dofs, see e.g. [4]) and underactuated (i.e. with less active cables than dofs, see e.g. [5]) topologies have been studied too. Redundant cable robots are the sole robots which can completely restrain the moving platform of a cable robot: in order to fully constrain the moving platform of a cable robot, it is required that the number of cables is greater by one than the number of dofs of the moving platform (see e.g. [6]). A higher number of cables may lead to overconstrained configurations (see e.g. [2]), while a lower number of cables leads necessarily to underconstrained robots, which must rely on gravity to keep positive tensions in the cables (see e.g.[7]). Indeed, contrary to fully constrained
or overconstrained cable robots, the underconstrained ones cannot take advantage of redundant cables to set a desired tension distribution in the cables. This makes operating underactuated cable robots particularly challenging.

A major requirement that has to be met in cable-robots is ensuring that during operation all the cables are under adequate tension (at least cable slackness must be prevented in all the cables), and that such a tension is below the maximum permissible value related to the torque limits of the winch motors or to the tensile force limits of the cables [4]. In practice, this makes it necessary to take into account explicitly the bilateral bounds on cable tensions reflecting both the unilateral nature of cables as actuators (cables can pull but are unable to push the end-effector) and the additional constraints posed, on the upper bound, by cable and motor physical properties and, on the lower bound, by safety margins or end-effector stiffness requirements [8]. The latter requirements usually suggest imposing a lower bound for cables forces greater than 0. Clearly, the evaluation of the performances of a cable robot cannot neglect such peculiarities of cable robots and the complexity arising from the need of keeping bounded cable tensions. As a result, though cable driven robots are basically parallel robots, the performance indices typically suggested for parallel robots (see e.g. [9]) cannot be employed straightforwardly.

So far, just a few examples of performance indices for cable robots have been proposed in literature. They have mainly been conceived as extensions to cable robots of traditional Jacobian-based performance indices. In [10] an evolution of the Yoshikawa manipulability has been proposed. In [11] the condition number has instead been applied as is, by restricting the analysis to a specific workspace. An evolution of the isotropy index, called tension factor, has been proposed in [12]: the tension factor is an isotropy index, defined in the joint space, which evaluates the ratio between minimum and maximum cable tensions. Another isotropy index has been defined in [7] for investigating the inertial properties of two rehabilitation cable robots. In the same work, a maximum isotropic force has been defined in order to find the minimum force that can be exerted in any direction. The index in [7] has been then extended in [13] for application to reconfigurable cable robots with one or more moving pulley blocks.

In [14] a novel approach to cable robot performance evaluation has been proposed and applied to solely redundant cable robots. The approach is based on the computation of the maximum force which can be exerted by the active cables on the moving platform along a specific direction. By extending the reasoning behind such an approach, this paper introduces a novel performance index called Wrench Exertion Capability (WEC) which can be applied to any cable robot topology. The reason for referring the evaluation to a given direction comes from a typical practical need when designing a cable robot: predicting the maximum force or torque that can be exerted on the moving platform along a direction of interest, usually keeping either null or limited wrench components, both in terms of forces and torques, along the other directions. This is basically what we mean by evaluation of the WEC of a cable robot along a direction.

Not only can the WEC be employed for maximum force/torque evaluations, but also to compute the minimum force/torque values which can be guaranteed throughout the workspace, irrespective of the direction, and for isotropy evaluations. All these evaluations allow getting considerable insight into cable robot performances and give the possibility to perform comprehensive comparisons among the performances of cable robots with different topologies and cable layouts: it is apparent that not only are the performances of cable robots influenced by the number of active cables, but also by their geometrical arrangement.

The paper is organized as follows. The WEC formulation is developed in Section 2: the formulation is based on the theory developed in [14] and here extended to underactuated robots: a general formulation is inferred. In Section 3 illustrative examples of computation of the WEC are provided: firstly, the WEC is employed to compare the performances of the two fully actuated planar cable robots with different cable layouts. A comparison is also made with state-of-the-art performance indices. Successively, an investigation is proposed to show the benefits of increasing the number of active cables attached to the moving platform of an underconstrained cable robot. The conclusions are stated in Section 4.
2 WRENCH EXERTION CAPABILITY

Suppose that for a given pose of the moving platform of a cable robot you were interested in evaluating the maximum force or torque that cables can exert on the platform, along a certain direction. This interest could be motivated by a variety of reasons, including for example the need to identify the regions of the workspace where the robot best performs in terms of initial acceleration, payload capacity or capability to react to external disturbances (forces or torques). Clearly, the maximum force or torque that the cables can exert on the moving platform, along a given direction, depends on the maximum force that each cable can exert. Less obviously, such wrench exertion capability also depends on the minimum tension of the cables that must be guaranteed to avoid cable slackness or to meet a desired stiffness requirement for the robot platform. At least, such a minimum force must be greater than zero so as to guarantee that cable forces can be maintained tensile. Computing the WEC index for a cable robot basically consists in performing the aforementioned evaluation taking into account cable tension limits explicitly.

The computation of the WEC suggested in this work is based on the solution of a linear programming problem involving cable tensions, cable tension limits, and a novel representation of the so-called wrench matrix. The wrench matrix (or structure matrix) $S$ of a cable robot usually defines the relation between the wrench $w_c$ exerted by the cable forces on the moving platform and the tension vector $\tau$ containing the cable forces $\tau_i$ (see Figure 1). It can be immediately recognized that in the most general case of a spatial cable robot driven by $m$ cables, it holds: $w_c = S\tau$, where the structure matrix $S$ takes the following form:

$$ S = [u_1 \times r_1 \ u_2 \times r_2 \ldots \ u_m \times r_m] $$

As schematically shown in Figure 1, vectors $u_i$ and $r_i$ are respectively the $i^{th}$-cable unit vector, (oriented from the moving platform, i.e. the box painted in gray in the scheme, towards the cable output point on the fixed frame) and the vector from the center of mass (G) of the moving platform to the point where the $i^{th}$-cable is connected to the moving platform.

The structure matrix $S$ only allows computing the cable wrench $w_c$ exerted by the cables on the moving platform. In general, this is not the sole wrench applied to the moving platform. In order to compute the total wrench $w = [f^T \ t^T]^T$ applied to the moving platform, external loading, including, for example, gravity force, should be taken into account. In the previous definition of $w$ vectors $f$ and $t$ are respectively the overall forces and torques exerted on the moving platform by the cables and the external forces. In order to account explicitly for external forces, a novel definition for the wrench matrix (denoted by $W$) is introduced, which is obtained by simply aggregating the structure matrix $S$ and the external wrench $w_e$:

$$ w = w_c + w_e = S\tau + w_e = [S \ w_e]^{(T)} = W^{(T)} $$

Once the matrix definition of $w$ in Equation (2) is introduced, it is possible to develop cable robot performance analysis following a well established approach. In particular, in the performance analysis of parallel manipulators, it has been proved convenient to split Jacobian matrices into their "translational" and "rotational" parts [9] in order to evaluate independently
the translational and rotational capabilities of the robots. By applying the same idea to the novel
definition of wrench matrix $\mathbf{W}$ of a cable robot, it is here suggested to split it into two parts,

namely $\mathbf{W}_f$ and $\mathbf{W}_t$ (where $\mathbf{W} := \begin{bmatrix} \mathbf{W}_f^T & \mathbf{W}_t^T \end{bmatrix}^T$) to analyze separately force and torque exertion capabilities.

Such an analysis is particularly useful when it is referred to a specific direction of interest. Since
the force and torque components in vector $\mathbf{w}$ are expressed in an absolute reference frame, in
order to refer the evaluation to a specific direction $d$, a rotation matrix $\mathbf{R}$ can be introduced to
define such a direction of interest univocally in the absolute reference frame adopted [15]. Once
the direction $d$ is defined, symbols $o1$ and $o2$ can be used to denote orthogonal Cartesian
directions, and the following expressions can be adopted to rotate matrices $\mathbf{W}_f$ and $\mathbf{W}_t$:

$$
\begin{align*}
\mathbf{R}_f^T \mathbf{W}_f & := \begin{bmatrix} \mathbf{W}_{f_d} \\
\mathbf{W}_{f_{o1}} \\
\mathbf{W}_{f_{o2}} \end{bmatrix}, \\
\mathbf{R}_t^T \mathbf{W}_t & := \begin{bmatrix} \mathbf{W}_{t_d} \\
\mathbf{W}_{t_{o1}} \\
\mathbf{W}_{t_{o2}} \end{bmatrix}
\end{align*}
$$

Then, for example, the WEC of a fully constrained cable robot can be expressed in terms of the
maximum force $w_{f_d}$ that can be exerted along the direction $d$ while keeping bounded cable
tensions and given values $\bar{w}_R$ of the other wrench components along the directions $d$, $o1$, and
$o2$. Such a WEC may be referred to as $\text{WEC}_f$, since it involves a force evaluation along a
direction of interest $d$, and can be computed by solving the following linear programming
problem (henceforth, the symbol $\preceq$ stands for the componentwise inequality):

$$
\begin{align*}
\text{WEC}_f := \max \left( w_{f_d} = \mathbf{W}_f \begin{bmatrix} \tau \end{bmatrix} \right) \quad \text{s. t.:} \\
\begin{bmatrix} \mathbf{W}_{f_{o1}} \\
\mathbf{W}_{f_{o2}} \\
\mathbf{W}_{t_{o1}} \\
\mathbf{W}_{t_{o2}} \end{bmatrix} \begin{bmatrix} \tau \\
\end{bmatrix} & := \mathbf{A} \begin{bmatrix} \tau \end{bmatrix} = \bar{w}_R \\
\tau_{\min} \preceq \tau & \preceq \tau_{\max}
\end{align*}
$$

On the other hand, if the torque exertion capability along a direction $d$ is to be evaluated, the
$\text{WEC}_t$ can be computed by solving the linear programming problem stated as follows:

$$
\begin{align*}
\text{WEC}_t := \max \left( w_{t_d} = \mathbf{W}_t \begin{bmatrix} \tau \end{bmatrix} \right) \quad \text{s. t.:} \\
\begin{bmatrix} \mathbf{W}_{f_d} \\
\mathbf{W}_{f_{o1}} \\
\mathbf{W}_{f_{o2}} \\
\mathbf{W}_{t_{o1}} \\
\mathbf{W}_{t_{o2}} \end{bmatrix} \begin{bmatrix} \tau \\
\end{bmatrix} & := \mathbf{A} \begin{bmatrix} \tau \end{bmatrix} = \bar{w}_R \\
\tau_{\min} \preceq \tau & \preceq \tau_{\max}
\end{align*}
$$

In general, a default value for $\bar{w}_R$ should be $0$ since a primary interest is predicting the
maximum force or torque that can be exerted along a direction keeping null generalized forces
(i.e. wrench components both in terms of forces and torques) along the other directions. This is
coherent with the typical investigation objectives presented at the beginning of this section.

The possibility of meeting imposed requirements on the full set of values $\bar{w}_R$ of the wrench
components excluding the one which is maximized, may only be assured in fully actuated and
redundant cable robots. Conversely, when a cable robot is underactuated, such a possibility is,
in general, prevented. Nonetheless, a suitable redefinition of the linear programming problem
allows extending the application of the WEC index to such cable robots. Indeed, in
underactuated cable robots it is impossible to apply the proposed optimization unless a
sufficient number of equations in the linear problem $\mathbf{A} \begin{bmatrix} \tau \end{bmatrix}^T = \bar{w}_R$ is removed. This is a
consequence of the fact that it is impossible to assign finite values to all the $\bar{w}_R$ components, but
clearly only to $m - 1$ of them, where $m$ is the number of active cables.

In practice, if finite values cannot be assigned to some wrench components, at least, it may be
useful to bound such components. Following this reasoning, a redefinition of the approach is
proposed, which consists in bounding the components to which finite values cannot be assigned. As an example, consider the problem statement in Equation (6) which refers to a spatial cable robot with 6 dofs driven by 3 active cables: in order to compute a \( \mathbf{W}_{f_d} \), finite values are assigned only to 2 force components \( \mathbf{w}_R \), while the torque components of the overall wrench are limited by upper and lower bounds \( \mathbf{w}_B \).

In particular, if we compare Equation (6) with the formulation in Equation (4), it is apparent that the constraints in the form of equalities removed from \( \mathbf{A}_1 \{ \tau \} = \mathbf{w}_R \) have been replaced by a suitable set of constraints in the form of inequalities \( \mathbf{B}_d \{ \tau \} \leq \mathbf{w}_B \).

Such an extension of the WEC definition to underactuated cable robots addresses a more general problem and gives the opportunity to make comparisons among considerably different robot topologies keeping a practical and functional approach. In Equation (7) a conclusive and general formulation of the WEC is given, where \( \Gamma \) is a generalized force (i.e. \( \Gamma \) can be either a force or torque component of the wrench vector \( \mathbf{w} \) projected along the direction \( d \) of interest).

It worth highlighting that in Equation (7) the dimensions of the matrices \( \mathbf{A} \) and \( \mathbf{B} \) are related to the cable robot topology and to the specific constraints defined for the analysis, reflecting operational requirements or specific features of the application. Let \( k \) be the number of rows of matrix \( \mathbf{A} \) (i.e. the number of linear equality constraints). The following inequality must always hold: \( k \leq \min (n - 1, m - 1) \), where \( n \) is the number of degrees of freedom of the moving platform and \( m \) is the number of active cables. The inequality holding for \( k \) reflects the practical need that the maximum number of equality constraints cannot exceed \( n - 1 \) (being 1 the force or torque to be maximized and \( n \) the overall number of wrench components) or \( m - 1 \) in the case of underactuated robots. The maximum number of rows \( l \) of matrix \( \mathbf{B} \) is instead equal to \( n - 1 - k \): since one inequality constraint can be introduced only if it is impossible, or it is not necessary for the given application, to provide \( n - 1 \) constraints in equality form (i.e. if \( k < n - 1 \), then \( l \) can be greater than zero). As for the number of columns of \( \mathbf{A} \) and \( \mathbf{B} \), they are both equal to \( m + 1 \), i.e. the size of the column vector \( \{ \tau \}^T \).

In conclusion the WEC definition is strictly related to the constraints defined for the linear programming problem. Such constraints depend on the topology and on the application. It is important to notice that the formulation with inequality constraints can be always adopted, while exact values to all the wrench components apart from the one maximized (i.e. a formulation with a full set of \( n - 1 \) equality constraints) can be imposed only with fully actuated or redundant robot.
3 METHOD APPLICATION

The WEC computation presented in the previous section is here applied to the performance analysis of the two groups of planar cable robots schematically depicted in Figure 2. The objective is to provide a clear proof of the effectiveness and usefulness of the WEC when it is employed at the design stage to find the regions of the workspace where a cable robot can best perform, or to compare different cable robot topologies or cable layouts. In particular, two representative investigations are carried out, addressing performance changes when:

- a) robots have identical topologies but different cable layouts,
- b) a cable robot topology is altered by increasing the number of active cables.

![Figure 2. The two groups of cable robots investigated](image)

The first group of cable robots investigated is shown in Figure 2 (a) and comprises two fully constrained, 3-dof and redundant cable robots which only differ in cable layout. The moving platform of the robots is rectangular (0.4 m x 0.2 m) and is driven by four cables attached to the four platform vertices. The cable output points are located at the four vertices of a square. The coordinates \((x, z)\) of the cable output points, expressed in a reference frame located at the square centroid, are: A \((-1 m, -1 m)\), B \((1 m, -1 m)\), C \((1 m, 1 m)\), and D \((-1 m, 1 m)\). It is assumed that the platform moves in a horizontal plane (plane \(xz\)), and that the platform, drive pulley, and winch designs allow avoiding cable interference in the robot with crossed cables (Figure 2 (a), on the right). For both the robots the WEC can be computed by employing the formulations proposed in Equations (4) and (5), since both the robots are redundant, and the platform can be fully constrained. In this example, it has been chosen to impose null wrench components \(\mathbf{w}_R = \mathbf{0}\) in the directions orthogonal to one along which a force or torque is maximized. Additionally, since no external wrench is assumed to be applied on the platform, \(\mathbf{w}_e\) is a null vector too. The maximization of all the three wrench components along the Cartesian axes \(x, y, z\) has been investigated, i.e. two forces acting along the positive directions of the axes \(x\) and \(z\), and a torque about the positive direction of axis \(y\). For example, the WEC, in terms of maximum torque about the positive direction of axis \(y\), has been computed as follows:

\[
WEC_{y} = \max \left( w_{ty} = W_{ty} \{\tau\} \right) \quad \text{s. t.: } \begin{cases} \{W_{fx}\} \{\tau\} = A \{\tau\} = \{0\} \\ \tau_{min} \leq \tau \leq \tau_{max} \end{cases}
\]

Figure 3 collects the results achieved. In Figure 3 (a), the sketches of the two cable robots can be recognized: dash-dotted lines are employed to connect the four cable output points; cables are represented by blue lines, and the moving platform is depicted in solid black line. A green quadrangle delimits the Static Equilibrium Workspace (i.e. the set of moving platform poses for which static equilibrium can be obtained while maintaining positive tensions in all the cables).
The SEW has been computed with the shown orientation of the platform, i.e. with the sides parallel to the $x$ and $z$ axes. As an example, the WEC has been evaluated at point $P(-0.2 \, m, 0.2 \, m)$, and refers to the force exertion capability along the positive direction of axis $x$. As far as the range of tensions that can be resisted by the cables is concerned, without loss of generality, the maximum value has been set to $100 \, N$ while the minimum to $5 \, N$. The arrows in
bold line overlapped to the cables provide a scale representation of the cable forces which allow achieving the maximum force represented by the red arrow, whose module is the $WEC_x^f$ value, in N, computed at point $P$, which is also written above the red arrow.

While the results shown in Figure 3 (a) refer to a single point $P$, the other plots of Figure 3 extend the analysis to the whole SEW: WEC values have been computed only for the points belonging to the SEW and without altering the moving platform orientation. Figure 3 (b) shows the values ($N$), taken by the $WEC_x^f$, while Figure 3 (c) shows the $WEC_z^f$ ($N$). Finally, Figure 3 (d) addresses the torque exertion capabilities of the robots by showing the $WEC_y^f$ (Nm). In all the subplots from (b) to (d) WEC values are represented by isolines. The regions where the best performances are achieved can be immediately recognized: they are the ones where the isolines take the highest values (i.e. those where they are painted in red). The comparison of the WEC values clearly highlights the superior performances that can be guaranteed by the robot with crossed cables. First of all such a robot, which also has a wider SEW, guarantees the possibility of exerting high forces in the $x$ and $z$ directions in a wider subset of the SEW (notice the extension of the red and orange isolines in subplots (b) and (c)). In terms of very maximum values taken by the forces, there are no significant differences between the two robots. Conversely, the torque exertion capability is completely different (see subplot (d)): the behavior of the robot with crossed cables is preferable since its cables can exert much higher torques.

The WEC formulation can be further exploited to compute the minimum force values which can be guaranteed along any direction and at any point of the SEW, and to perform an isotropy evaluation based on comparing such minimum force values with the very maximum ones that can be exerted at any point of the SEW. As an example, such analyses are presented in Figure 4 with reference to the sole robot with crossed cables. Figure 4 (a) represents such a robot at point $P(-0.3m, -0.3m)$, the result of a $WEC_d^f$ computation performed along any direction $d$ is depicted through a red polygon which provides a scale representation of the maximum force that can be exerted along any the radial direction from $P$. An angular resolution by 1° has been adopted to trace such a polygon. The wrench exerting capabilities of the robot referred to all the possible directions can be immediately inferred. The maximum and minimum exertable forces can be easily found too: they are traced in green lines.

If the same $WEC_d^f$ computation (i.e. repeated along any direction) is performed at each point of the SEW, a minimum force value which can be guaranteed irrespective of the direction can be found. Figure 4 (b) shows such a value plotted through isolines. Getting such information at the design stage is obviously of great practical usefulness.

If, at any point, the ratio between the minimum and maximum exertable forces (e.g. those traced in green lines for point $P$ in the subplot (a)) is computed, an effective isotropy evaluation can be carried out. Figure 4 (c) shows such ratios, which can be compared to the popular isotropy index called Tension Factor (TF), proposed in [12]. The TF is the ratio between the minimum and the maximum cable tension values achieved when the platform is in static equilibrium. In Figure 4 (d) the TF computed at each point of the SEW has been plotted by isolines. Apparently, subplots (c) and (d) provide different indications in terms of robot isotropy, however it is the authors' opinion that the one based on the WEC is more useful in practice, since the TF provides a measure of robot isotropy in the joint space, and hence, it does not provide straightforward information on the performance in the Cartesian space.

The second group of cable robots investigated is shown in Figure 2 (b) and comprises four underconstrained (or "cable suspended"), 3-dof cable robots which differ in the number of cables and/or in the cable layout. The robots are assumed to move in a vertical plane, hence, their platforms are under the influence of gravity, which is essential to maintain tension in the cables in static conditions. Once again, the plane of motion is plane $xz$. The moving platforms of these robots are identical to one another and have the same shape and dimensions of the platforms of the previous group of robots.
Figure 4. Scale representation of the maximum force that can be exerted along any direction at point \( P \) (a), minimum guaranteed force (b), isotropy evaluation (c) and Tension Factor (d).

The cables are attached to the platform upper vertices. The cables output points are instead located at the two upper vertices of a square. The coordinates \( (x, z) \) of such points, expressed in a reference frame located at the square centroid, are: A \((-1 \ m, 1 \ m)\) and B \((1 \ m, 1 \ m)\). The first robot only has two cables, and hence it is also underactuated. The other robots are instead fully actuated or redundant (i.e. with four cables).

By comparing the performances of these robots, the effect of increasing the number of active cables can be appreciated, also in relation to the cable layout adopted. This is the objective of such a comparative analysis, whose results are collected in Figure 5. Robot performances are compared by referring to the \( WEC^f_{cg} \) obtained by assuming that cable tensions are to be kept in the range 5-100 \( N \), that a null overall torque must be exerted on the platform, and that a limited vertical force in the upwards direction has to be applied to the platform. The lower and upper bounds of such a vertical force have been set equal, respectively, to 0 and 5 \( N \). The mentioned constraints (i.e. null torque and limited upwards vertical force on the platform) are coherent with the field of application where cable suspended robots are very likely to be employed in the future: high speed pick and place manipulations (e.g. over-the-belt packaging). Indeed, if a cable suspended robot has to be employed for such tasks, it is of apparent interest evaluating which is the maximum horizontal force that can be exerted on the platform, and hence applied to the picked object, while keeping a null torque on it (not to induce rotations) and (if a given force value in the vertical direction cannot be set due to the limited number of cables available) a bounded upward force. Not only does imposing an upward force meet the basic requirement of lifting the picked objects during the manipulation, but setting an upper bound could allow preventing the load from being dropped at the start of the motion. Alternatively, the upper bound on the vertical force could reflect a limitation to the maximum vertical acceleration.
Figure 5. $\text{WEC}_f$ of different topologies of cable suspended robots presented in Figure 2 (b). $\text{WEC}_f$ are either computed at a single point $P$ (plots on the left) or throughout the SFW (plots on the right).
In the comparative analysis, though only the 2-cable robot is underactuated, the same constraints in the form of inequality are kept for all WEC computations in order to make the comparison between different cable robot topologies fair. The problem is stated as follows:

\[ WEC_x^f := \max \left( w_{fx} = W_{fx} \begin{bmatrix} 1 \end{bmatrix} \right) \quad \text{s.t.:} \quad \begin{bmatrix} W_{fy} \\ W_{fx} \\ -W_{fy} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = 0, \quad \begin{bmatrix} B \\ -B \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \leq \vec{w}_B \]

The \( WEC_x^f \) can be computed at any point achievable by the moving platform statically or dynamically. In the subplots on the left of Figure 5, the computation is referred to a generic point \( P(-0.2m, 0.2m) \). The lines employed in these plots have the same meaning of the corresponding ones in Figure 4 (a). Here, however, a vertical green arrow is also adopted to provide a scale representation of the external wrench \( \vec{w}_e \): the force of gravity acting on the platform. The mass of the platform has been set equal to 5 kg.

In the subplots on the right of Figure 5, the analysis has been extended to all the points of the Statically Feasible Workspace (SFW) defined as the set of the mobile platform poses for which static equilibrium against gravity can be obtained using a limited range of cable tensions [3]. The SFWs of all the robots, computed keeping the platform horizontal, are represented in the subplots on the left of Figure 5, for clarity, delimited by green solid lines. The SEWs have been geometrically bounded by the square box with vertices at A and B. The isolines in the plots highlight the greatly different behaviors of the four cable driven robots: while for the underactuated robot (subplot (a)) it is possible to find a solution to the problem stated in Equation (9) in just one point of the SFW, by increasing the number of active cables, the force exertion capabilities improve considerably. The subplots (b) and (c) prove that, for the given problem of maximizing a rightward force, the two fully actuated robots behave very differently in their SFW. The SFWs of the robots are very different (basically symmetrical about the z axis) and not overlapped, which complicates performing a straightforward comparison between the robots. Nonetheless, the plots provide clear hints about the regions where these robots can best perform. Clearly, the most effective cable layout could be identified once the geometrical features of the tasks to be executed and of the workcell were known: in general, the cable arrangement of Figure 5 (b) seems preferable since the \( WEC_x^f \) takes high values in a wider region of the workspace. Finally, in Figure 5 (d), the \( WEC_x^f \) of the redundant robot is plotted. Obviously, the availability of a fourth cable allows extending the SFW and improving the performances within it. In particular, the rightward force exertion capability shows that this robot merges the benefits of the fully actuated robots discussed earlier, at the expense of an increased cost, design complexity, and cable obstruction in the workspace.

4 CONCLUSIONS

This paper has introduced a novel performance index named Wrench Exertion Capability (WEC), which has been suitably developed for cable driven robots. The WEC allows evaluating the maximum force or torque exertion capabilities of a cable robot along a direction of interest, and accounts explicitly for the intrinsic cable tension limits and for the constraints which can be imposed to the wrench components that are not maximized. A linear programming problem is solved to compute the WEC. The problem makes use of suitable partitions of a novel definition of wrench matrix which has been introduced to simplify the inclusion of external wrenches in the analysis. The WEC formulation proposed is general enough to allow the analysis of redundant, fully actuated and underactuated cable robots. To this purpose, the use of constraints in both the form of equality and inequality has been suggested and discussed. Finally, representative examples of performance comparisons among different robot designs by the WEC have been presented.
<table>
<thead>
<tr>
<th>REFERENCES</th>
</tr>
</thead>
</table>
Reduction of the effect of actuator saturation with periodic servo-constraints

László Bencsik, Ambrus Zelei, László L. Kovács

MTA-BME Research Group on Dynamics of Machines and Vehicles Budapest, H-1111, Hungary [bencsik, kovacs, zelei]@mm.bme.hu

ABSTRACT
The use of servo-constraints, or program definition in the task-space is quite common in trajectory tracking control of manipulators. The task definition is straightforward in case of fully actuated systems. Several algorithms providing the desired motion of the whole system can be found in the related literature. An originally fully actuated system can be handled as underactuated when some of the actuators reach their torque limits. In case of underactuated systems, the task is also given. Based on the task the controlled and uncontrolled directions can be separated. The uncontrolled motion, also referred as the internal dynamics of the system, has to be stable to ensure the stability of the whole system. Thus, the stability of the internal dynamics depend on the output represented by the servo-constraint. It means that in order to get a realizable task, the behaviour of the internal dynamics should be considered, and in many cases the original task has to be slightly modified. This paper presents a novel approach, when the servo-constraints are used for handling actuator saturation and not simply modified, but also switched periodically in time. It will be shown that the application of periodic servo-constraints is decreasing the trajectory tracking error in case of actuator saturation.

1 INTRODUCTION
Working with the control of underactuated mechanical systems can be inspired by several practical and theoretical problems. Beyond the classical underactuated problems, like cranes, flexible structures and aerial vehicles, the actuator saturation also can be mentioned as a characteristic underactuation [1].

Because of the wide variety of underactuated systems there is no general recipe for the control of them, but some metrics can be found - like the relative degree or the flatness of the system - to categorize them and find an appropriate control algorithm based on that categories. The mechanical modelling of underactuated systems as special type of complex multibody systems are well developed [2, 3]. Most of these rigid body dynamics based approaches use constraint equations in the mathematical model. On one hand, geometric or kinematic constraints give the relation of the dependent descriptor coordinates. Additional geometric constraints defined by the user are responsible for the task definition and therefore they are called servo-constraints. While the geometric or kinematic constraints are naturally satisfied, in some cases the servo-constraints are cannot be fulfilled by different reasons. Several publications [3, 4] deal with the modification of the original servo-constraints in order to get a realizable task. An other possible approach is the periodic variation of the servo-constraint. For underactuated robots the usefulness of the method was proven in [5].

In case of industrial robotic applications, when the trajectory design was careful and operating conditions are well defined in most cases the actuator saturation can be avoided, but always there will be a trade-off between selecting saturation preventing operational conditions and productivity. The most used practical technique to avoid the unwanted effect of actuator saturation is the so-called anti-windup scheme [6], when the original controller is subjected to a compensator which takes into account the difference between the saturated and ideal control inputs. In [1], the saturation of
actuator(s) are handled as a temporary reduction of the number of available independent control inputs. In that work during the saturation the number of prescribed servo-constraints is reduced in order to get a realizable task. In contrast, present paper introduces the further generalization of the periodic variation of servo-constraints [5] in case of actuator saturation, so the servo-constraints are systematically switched during actuator saturation to redistribute the load on the actuators. When the actuators are not saturated, a general computed torque control scheme is applied to realize the desired motion.

2 PROBLEM FORMULATION

Most of the controlled mechanical systems possess complex multibody structure of which the mathematical modeling is convenient by using non-minimum set of descriptor coordinates. In such case, the general form of the equation of motion is written as

\[
M \ddot{q} + \Phi \dot{q} \lambda = Q + H \dot{u},
\]

\[
\phi = 0,
\]

where \( q \in \mathbb{R}^k \) are dependent coordinates, \( M(q) \in \mathbb{R}^{k \times k} \) is the mass matrix, \( \Phi(q,t) \in \mathbb{R}^{m \times k} \) is the Jacobian of geometric constraints \( \phi_g(q,t) \in \mathbb{R}^m \) and \( \lambda \in \mathbb{R}^m \) is the vector of the Lagrangian multipliers which is related to the magnitude of the constraint forces and torques. Matrix \( H(q) \in \mathbb{R}^{k \times l} \) is the control input matrix and \( u \in \mathbb{R}^l \) contains the actuating forces and torques. In addition, \( Q(q,\dot{q},t) \in \mathbb{R}^k \) denotes the remaining generalized forces.

The equation of motion (1) with the geometric constraints (2) forms a differential algebraic equation (DAE) with differentiation index 3. Using the method of Lagrange multipliers [7] the constraints are considered on the level of acceleration:

\[
\ddot{\phi}_g = \Phi q \ddot{q} + \dot{\Phi} q \dot{q} + \dot{\phi}_g t,
\]

where

\[
\Phi_q = \frac{\partial \phi_g}{\partial q}, \quad \phi_{gt} = \frac{\partial \phi_g}{\partial t}.
\]

With the above index reduction technique the original DAE problem can be solved as an ordinary differential equation (ODE).

2.1 Task definition by means of servo-constraints

In fully actuated case the \( l \) number of independent actuators is equal to the \( n = k - m \) degrees of freedom. In case of actuator saturation the \( l \) number of independent actuators becomes less than \( n \). In such case the system is called underactuated.

The required motion is specified by the so-called servo-constraints [11, 2] \( \phi_s(q,t) \in \mathbb{R}^l \). The additional constraint equations

\[
\phi_s = 0,
\]

have a mathematical form similar to the geometric constraints. With the loss of generality we suppose that the relative degree for all outputs are \( r = 2 \) [13]. If this condition is not satisfied by the servo-constraints, further generalization is possible to be carried out. The physical meaning of \( r = 2 \) is that the input forces or torques have a direct effect on the system in the controlled directions. Mathematically it means that the differentiation index is still 3 when the system (1) and (2) are subjected to the servo-constraint equation (4). Thus using the method of Lagrange multipliers the servo-constraints also can be considered on the level of acceleration as

\[
\dot{\phi}_s = G q \ddot{q} + \dot{G} q \dot{q} + \dot{c}.
\]
where
\[ G_q = \frac{\partial \phi_s}{\partial q}, \quad c = \frac{\partial \phi_s}{\partial t}, \]
similarly to the geometric ones. In case of the servo-constraints the control input \( u \) plays a similar role like the Lagrangian multipliers of the geometric constraints.

### 2.2 Computed torque control method

During the control design of mechanical systems it is often expedient to calculate the inverse dynamics of a system with respect to the desired task. The solution of the inverse dynamics can be seen as a feedforward control action that realizes the desired motion without considering any disturbances and modelling errors. This approach is frequently referred in the literature as computed torque control method. In case of underactuated systems the inverse dynamics is not well defined. Some degrees-of-freedoms cannot directly be controlled, and the corresponding generalized coordinates depend on the system dynamics only. Still, the input forces can be computed from the following equation constructed by using equations (1), (3) and (5) [4].

\[
\begin{bmatrix}
M & \Phi_q & -H \\
\Phi_q & 0 & 0 \\
G_q & 0 & 0 \\
\end{bmatrix}
\begin{bmatrix}
\ddot{q} \\
\lambda \\
\dot{u} \\
\end{bmatrix}
= 
\begin{bmatrix}
Q \\
-\Phi_q q - \dot{\phi}_g \\
-G_q q - c - K_P \dot{\phi}_j - K_P \dot{\phi}_j \\
\end{bmatrix},
\]

(6)

were the control gains \( K_P \) and \( K_D \) are the proportional and derivative gains have similar role like the parameters of the Baumgarte stabilization technique [12] which is quite common for stabilizing geometric constraints in multibody simulations. We use (6) in the case of the original fully actuated system and also in the underactuated case, when some of the actuators temporarily saturate.

### 2.3 The periodic servo-constraints

As it is presented in the previous section the task of the manipulator can be formulated as additional constraints called servo-constraints. Many techniques can be found in the literature to solve the resulting DAE problem. In reference [4] and in [2] the original task thus the servo-constraints are slightly modified in order to stabilize the internal dynamics or get a feasible problem. This modification makes the stable control possible. Obviously, this modification has drawbacks and it results larger, but still acceptable, tracking errors. Reference [5] introduces a different approach when the servo-constraints are switched periodically in time. In one period the servo-constraints are responsible for realizing the desired motion, while in the subsequent, typically shorter period a different set of servo-constraints are formalized to stabilize the internal dynamics.

### 2.4 The issue of actuator saturation and frame algorithm

We focus on fully actuated manipulators with \( n \) degrees of freedom and \( l = n \) actuators performing a strictly \( l = n \) dimensional task. If an actuator reaches its torque limit than it will not be able to provide the required force. As a consequence, if \( m \) number of actuators saturate than the maximum dimension of the independently feasible task is reduced to \( l = n - m \).

The proposed method of periodic servo-constraints is divided into two main parts. In the case, when the actuators are not saturated, a simple computed torque control scheme is applied. During saturation new set(s) of servo-constraints are introduced and they are switched in time. The periodic servo-constraints requires a switching pattern introduced in the section 3. In order to determine the new set(s) of servo-constraints the reduced control input matrix \( \hat{H} \) should be derived. For the partitioning of the servo-constraints the relative degree analysis [13] gives us a hand. The relative degree \( r_{ij} \) should be analysed between the \( i \)th saturated actuators (\( i = 1, 2, ..., m \)) and the \( j \)th servo-constraints (\( j = 1, 2, ..., n \)). If \( r_{ij} > 2 \), then the \( i \)th saturated actuator has not effect on the \( j \)th constraint on the acceleration level. These servo-constraints are involved in every new sets in their original form. If \( r_{ij} = 2 \), then the saturated actuator has direct effect on the \( j \)th servo-constraint. In
this case we have to find an independent actuator among the \( g = 1, 2, \ldots, n - m \) unsaturated actuators which has also affect on the \( j^{th} \) servo-constraint, so that \( r_{gj} = 2 \). Hereupon the servo-constraints corresponding to \( g^{th} \) actuator are switched in order to achieve that \( l = n - m \) number of servo-constraints are used simultaneously. The proper selection of the switching pattern guarantee the minimization of the servo-constraints’ violations. Section 3 will introduce the simplest example on which the idea of periodic servo-constraint can be used to reduce the effect of actuator saturation.

3 SIMULATION CASE STUDY

In order to illustrate the applicability of the presented approach a simulation study was accomplished on a two-link manipulator shown in Fig. 1. The corresponding members of the equation of motion \((1)\) of the manipulator was derived from the Euler-Lagrange formula using the minimum set of generalized coordinates \( q = [\theta_1, \theta_2]^T \) (see Fig 1). The mass matrix is

\[
M = \begin{bmatrix}
J_0 + J_{CM2} + \frac{1}{4} l_2^2 m_2 + l_1 l_2 \cos \theta_2 & J_{CM2} + \frac{1}{2} l_1 l_2 m_2 \cos \theta_2 & J_{CM2} + \frac{1}{4} m_2 l_2^2 \\
J_{CM2} + \frac{1}{2} l_1 l_2 m_2 \cos \theta_2 & J_{CM2} + \frac{1}{4} m_2 l_2^2 & J_{CM2} + \frac{1}{4} m_2 l_2^2
\end{bmatrix}.
\]

The coefficient matrix of the actuator forces \( u = [\tau_1, \tau_2]^T \) can be derived from the virtual power of the actuators

\[
H = \begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix},
\]

while the remaining forces are

\[
Q = \begin{bmatrix}
l_1 l_2 m_2 \sin \theta_2 (\dot{\theta}_1 - \frac{1}{2} \dot{\theta}_2) \dot{\theta}_2 \\
\frac{1}{2} l_1 l_2 m_2 \sin \theta_2 \dot{\theta}_2 \dot{\theta}_2
\end{bmatrix}.
\]

The links are supposed as homogeneous rigid bodies, and the physical parameters can be found in Table 1. The manipulator was placed perpendicular to the gravity field, like a typical SCARA robot application, thus the effect of gravity was not present.

The tool center point (TCP) of the manipulator was commanded to follow the desired trajectory. Using the idea of servo-constraints the task was written as

\[
\phi_b = \begin{bmatrix}
\phi_{b1} \\
\phi_{b2}
\end{bmatrix} = \begin{bmatrix}
l_1 \cos(\theta_1) + l_2 \cos(\theta_1 + \theta_2) - x_D \\
l_1 \sin(\theta_1) + l_2 \sin(\theta_1 + \theta_2) - y_D
\end{bmatrix},
\]

where \( x_D, y_D \) describe the desired TCP position on horizontal plane. In order to understand the operation of the proposed controller only the shoulder actuator can saturate. During saturation the input matrix is reducing as \( \hat{H} = [0 \ 1]^T \) in the inverse dynamics calculation, and the saturated actuator force is applied as a constant force appearing in \( Q \).
<table>
<thead>
<tr>
<th>Description</th>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mass of the first arm</td>
<td>( m_1 )</td>
<td>0.2 [kg]</td>
</tr>
<tr>
<td>Mass of the second arm</td>
<td>( m_2 )</td>
<td>0.2 [kg]</td>
</tr>
<tr>
<td>Length of the first arm</td>
<td>( l_1 )</td>
<td>0.4 [m]</td>
</tr>
<tr>
<td>Length of the second arm</td>
<td>( l_2 )</td>
<td>0.4 [m]</td>
</tr>
<tr>
<td>Inertia of the first arm</td>
<td>( J_0 )</td>
<td>0.00107 ([kgm^2])</td>
</tr>
<tr>
<td>Inertia of the second arm</td>
<td>( J_{CM2} )</td>
<td>0.0027 ([kgm^2])</td>
</tr>
</tbody>
</table>

3.1 The tuning of the pattern of the periodic servo-constraints

The key question of the method is the switching pattern of the servo-constraints. This switching rule can be derived in several ways. The following method focuses on the previously introduced 2 degrees of freedom example. During the actuator saturation the unsaturated actuator has to produce different accelerations in different directions, while the effect and necessity of the actuator on the different servo-constraint directions is not equal. Thus, based on the ratio of necessity we can determine a pattern for the switched servo-constraints. In order to approximate this ratio the mass matrix \( M \) and \( \dot{H} \) should be transformed into the space of servo-constraints. The time differentiation of the servo-constraint equations gives the following formulae on the velocity level:

\[
\dot{\phi} = G_q \dot{q} + \mathbf{c},
\]

where \( G_q \) plays the role of a Jacobian matrix, thus the mass matrix in the space of the servo-constraints is

\[
W = G_q^{-T} M G_q^{-1}.
\]

The virtual power of the available actuators can be written as,

\[
\delta P = \delta \dot{\phi}^T G_q^{-T} \dot{H} u,
\]

from where control input matrix in the space of the servo-constraints is

\[
B = G_q^{-T} \dot{H}.
\]

If the TCP is controlled by servo-constraints, the above mentioned transformation has a same result as the operational space control [10]. A possible approximation of the effect and necessity of the remaining actuators can be estimated by dividing the desired inertial force \( \mathbf{w} \) by each columns of the operational space control input matrix \( \mathbf{B} \)

\[
z_i = w_i / B_{i,i}; \quad i = 1, 2...n.
\]

The elements in \( \mathbf{z} \) will show the influence of the remaining independent actuators on the servo-constraints. The desired inertial force \( \mathbf{w} \) is calculated by the product of the effective (operational space) mass matrix \( W \) and desired servo-constraint accelerations \( \mathbf{c} \) as:

\[
\mathbf{w} = W \mathbf{c}.
\]

The remaining centrifugal and Coriolis forces are neglected in this approximation. The periodic pattern is constructed along the trajectory based on ratio \( \mathbf{z} \).

In case of the presented example only one actuator can saturate, so \( m = 1 \) and the remaining actuator has effect on the both servo-constraints. Thus two servo-constraint are switched, while the matrix \( \mathbf{B} \) has only one column. The periodic pattern can be constructed before the simulation based on the desired quantities.
In the present example $z \in \mathbb{R}^2$. Based on that we split the time periods into two parts when for $i$ time steps the first servo-constraint is valid and for $k$ time steps the second servo-constraint is valid. The ratio of $i$ and $k$ are directly proportional with the ratio of the elements of $z$:

$$i = \text{int} \left( \frac{p}{|z_1| + |z_2|} \right),$$

where $p = i + k$ is size of the time period. We calculate this ratio for each time period along the desired trajectory, thus $i$ and $k$ can be different in every period. Function $\gamma$ realizes the periodic switching pattern, see Fig. 2. The periodic servo constraint is $\hat{\phi}_s = \gamma \phi_{s1} + (1 - \gamma) \phi_{s2}$. In the case of the two-link arm $x$ position of TCP was controlled for $i$ time steps and $y$ position was controlled for $k$ time steps.

### 3.2 Numerical results

In the numerical simulations the manipulator had to follow an arc of a circle shown in Fig. 3. The whole desired path in the plane, the initial configuration (marked with 0) and the end configuration (marked with 1) are presented on Fig. 4. As it is mentioned, the actuator in the first joint can saturate at the value of $|\tau_{1,\text{max}}| = 0.6$[Nm] and the torque limit of the second actuator is not considered.

The proportional and the derivative gains in (6) was set to $K_P = 40$ $K_d = 20$ respectively in every simulation scenarios.

In the first simulation scenario the effect of actuator saturation is not handled by periodic switching of the servo-constraints. The control torques was calculated based on equation (6) and Fig. 5 clearly shows that the first actuator is saturated. Fig. 6 presents that during the saturation the servo-constraint violation was significantly increased.
Figure 4. The desired path and the initial and end configuration

Figure 5. Simulation results - actuator torques
In the second simulation scenario the proposed control algorithm were implemented on test example. In the initialization phase the whole path was divided into time periods. Each time period built up by $p$ samplings and function $\gamma$ was constructed based on equation (17). The number of samplings is $p = 20$. During the actuator saturation the periodic servo-constraints generate a control force as it presented in Fig. 5. The corresponding violations of the servo-constraints are visible in Fig. 6. Simulation results clearly show that the violation of the servo-constraints are significantly smaller. Besides, the system gets out from the saturation a little bit faster, as Fig. 5 shows. As a marginal drawback, the computed torques has a periodic-like oscillation, which can be provided by an actuator with average dynamical properties.

In order to make a quantitative comparison, the norm of the servo-constraint violation was plotted on the same chart on Fig. 7. The maximum error was $|\phi_s| \approx 45\text{[mm]}$ when the actuator saturation is not handled. When the periodic servo-constraints were used, the maximum error of the servo-constraints was only $|\phi_s| \approx 30\text{[mm]}$. To get a comparable metric the Root-mean square (RMS) value of the norm of the servo-constraints was computed as:

$$\bar{|\phi_s|} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} |\phi_s(t_i)|^2}.$$  

When the original servo-constraints was used the RMS value was $\bar{\phi}_s = 0.0174$ however for periodic servo-constraints the RMS value decreased to $\bar{\phi}_s = 0.0122$. Consequently the proposed control algorithm was able to reduce the effect of actuator saturation with approximately 30%.

4 CONCLUSIONS

In this paper the extension of periodically switched servo-constraint was proposed for the control of saturated systems. The proposed controller was tested on a two-link manipulator case study application. The results showed that with the application of the introduced method the violation of servo-constraints was much more acceptable during the actuator saturation. Thus we can conclude that with the application of periodically switched servo-constraints the precision of trajectory tracking can be enhanced during actuator saturation. An optimal switch pattern was generated for the two link case study example. However the optimalization method for general case is possible within a further research.
REFERENCES


Sensitivity analysis on MIMBOT biped robot through parallel computing

Alejandro Bustos*, Giuseppe Carbone#, Higinio Rubio*, Marco Ceccarelli#, Juan C. García-Prada*

* Department of Mechanical Engineering
Universidad Carlos III de Madrid
Av. Universidad, 30, 28911, Leganés, Madrid, Spain
[albustos, hrubio, jcgprada]@ing.uc3m.es

# LARM: Laboratory of Robotics and Mechatronics
DiCEM-University of Cassino and South Latium
Via G. Di Biasio 43, Cassino (Frosinone), Italy
[carbone, ceccarelli]@unicas.it

ABSTRACT

In this paper, a biped walking robot based on classical mechanisms with advanced skills is presented. Topologies of MIMBOT biped walking robot are described. Kinematic and dynamic equations of the proposed leg mechanisms are formulated and implemented in a program designed for carrying out the corresponding simulations. This program is designed in such a way that it exploits the parallel capabilities of GPUs to accelerate the computation. Finally, results are also presented showing the advantages of using GPUs.

1 INTRODUCTION

Legged locomotion is more efficient, speedy and versatile than the one by track as wheeled vehicles when it operates on a rough terrain [1]. This is why this research field has attracted great interest from university, laboratories and companies, which have built a lot of walking robot prototypes in any kind of leg configuration in the last decades [2]. In order to make a biped robot walking anthropomorphic like, most of the leg mechanism are built with an anthropomorphic architecture. However, this strategy presents a series of drawbacks, the mechanism design is complex, requires a high number of motors and gear boxes, needs sophisticated control algorithms and electronics and suffers the “Back-driven” effect [3]. The use of proper mechanisms simplifies the robot operation. However, a very careful design is needed for this purpose; proper kinematic and dynamic models have to be developed.

Other approach to this research field is the emulation of the human gait using classical mechanisms of 1 DOF (Degree Of Freedom). By this way, biped robots walk in an anthropomorphic way at the time they minimize the energy required to make this action. Walking chairs developed by Takeda et al. [4] are a perfect example of this philosophy. Prof. Ceccarelli and his team at the Laboratory of Robotics and Mechatronic, LARM, have been working in a continued way in this line, firstly with the different evolutions of biped robot EP-War and lately with the designs of the legs of the low-cost humanoid CALUMA [5] and others legged walking robots using classical mechanisms [6], [7].

From the cooperation between the LARM and MAQLAB groups arises the biped walking robot PASIBOT, Figure 1, about which several mechanical studies have been presented[8], [9],[10]. This biped is a 1 DOF mechanical system based in the smart combination of classical mechanisms that widely reproduce the human walking. The wish to improve the skills of biped leads to the evolution into MIMBOT thanks to the addition of two small linear actuators in each hip. Kinematic and dynamic models have been developed, but these models require the solution of 179 equations with large computational costs. For this purpose parallel computing can be very useful.
The GPUs (graphics processing unit) have a series of characteristics (hundred of cores, fast access to onboard memory, high level of task parallelization, etc.) [11] that make them very attractive to solve iterative and/or big problems. However, it’s not until 2006, when NVIDIA launches its GPU cards with CUDA architecture, when this comes true. This new architecture, along with the SDKs and APIs available for programming GPU for general purposes (GPGPU), has brought the computation of any kind of scientific issues to a new level, particularly in terms of speed and problem size. Also it is a cheaper (the most advanced GPU costs about $4000, but it’s easy to find a reasonably GPU for about $500) and low-power-consumption (compared to traditional supercomputers) technology [12].

Many researchers around the world have realized the power and capabilities of the GPGPU. Today, a lot of work is being done to convert existing technologies and algorithms to “GPU applications”, enhancing their performance. As said before, parallel computing over GPUs has not only helped to increase speed, but has also made many difficult (or nearly impossible) computations, possible. There are several examples of GPGPU use covering any kind of scientific discipline: Negrut et al. have solved large multibody dynamic systems involving millions of frictional contacts and bilateral mechanical constraints [13], [14]; Garcia Blas et al. have applied the GPGPU to improve the performance of X-Ray computed tomography [15], or Andrew and Dingle have implemented a QR factorization algorithm on GPU [16].

In this paper, a reduced DOF leg mechanism with linkage architecture is proposed. Kinematic equations are formulated. The parametric mathematical model defining the robot's kinematics is implemented in C++. In this way, it is possible to exploit the parallel characteristics of the GPUs, accelerating the execution time of algorithms. Finally, a study of the results achieved will be carried out.

2 THE ATTACHED PROBLEM

The biped robot MIMBOT is a reduced DOF mechanical system. It is based on the smart combination of classical mechanisms to imitate the human walking. Two main mechanisms make up the robot: a Tchebychev mechanism gives the robot the basic trajectory of its gait (as it generates an almost straight line trajectory) and a pantograph mechanism reverses and amplifies this movement. The pantograph’s central point (point M) is fixed to the hip, being driving by the short side (point B) and obtaining the final foot trajectory on the long side (point A). Finally,
a stabilization system must be added to ensure that the foot will be parallel to the ground without additional actuation.

Two small linear actuators (one horizontal and one vertical) placed on each side of the robot’s hip gives the biped some mimetic skills. The rod of both actuators will be attached to the fixed point of pantograph mechanism (point M), because in this way the geometry of the mechanism is modified, and consequently, the gait (trajectory of point A) will change to get the desired mimetic skill. These linear actuators allows the robot to execute human skills like lengthening or shortening the gait, making turns, going up and down stairs or avoiding obstacles of certain height.

In the biped’s evolution they can be differentiated five steps, as shown in Figure 2. The first one corresponds to a simple scheme of the leg, with the two main mechanisms. The second one adds the stabilization system, and the third one adds also two linear actuators to confer the new skills. The fourth step modifies the stabilization system in such a way that the foot will be always parallel to the ground [7]. Finally, last stage adds the linear actuators to the model with the new stabilization system.

The search for an optimal kinematic topology and size synthesis that can properly mimic human biped walking implies hundred of simulations in order to achieve the best solution. This computing effort leads to huge computational costs. An estimation of the required time for this task yields a result that oscillates between two and four months. Additionally, the required memory for data storage is, just, unfeasible, as tens of terabytes are needed, which leads to an obvious memory overflow without reaching any result.

Some authors have been working in parallel optimization algorithms running over GPUs. Reported results [17] show that speedups in the order of 1/50 are possible for the best conditions and depending on the algorithm complexity. Beyond the numbers, these works shows the increases in speed are greater the bigger is the amount of data to manage. They also present a better management of memory resources than the original sequential algorithms. GPUs generally have much less memory than the host, which force the programmers to make a clever use of memory resources that generally results in a reduction of memory requirements.

Therefore, parallel computation using GPUs capabilities gives us the opportunity to reduce dramatically the time spent in the kinematic/dynamic computation, as well as in the search phase and the whole optimization process. In addition, programming the kinematic/dynamic and optimization algorithms for GPUs gives us a better management of the memory resources and the opportunity to avoid the above mentioned memory overflows. Doing these tasks in CUDA C++ environment also allows taking full advantage of the most recent features that have been developed for GPGPU.
3 ANALYTICAL FORMULATION

Posing and solving the closure equations corresponding to each kinematic chain that makes up the mechanism, it is possible to find the expressions that define the positions, velocities and accelerations of every link as a function of the position and angular velocity of the input crank \( \theta_2 \). Taking this into account, the angular positions of any link can be referred as follows:

\[
\theta_i = \theta_j(\theta_2), \quad i = 1, 2, \ldots, 1', 2', \ldots
\]  

Then, the \( x, y \) coordinates for its centre of mass, can be easily expressed with respect to that angle:

\[
X_i^{CDM} = X_j^{CDM}(\theta_2); \quad Y_i^{CDM} = Y_j^{CDM}(\theta_2), \quad i = 1, 2, \ldots, 1', 2', \ldots
\]  

Furthermore, if the time dependent function for the input crank is known, these equations can be expressed as time dependent functions, and then, the velocities and accelerations obtained by taking their first and second time derivatives.

The biped kinematics is based on 4 closed loop kinematic chains, corresponding to the Tchebychev, pantograph and upper and lower sections of the stabilization system. Using the Raven method it is possible to write the required equation systems that solve the kinematic problem:

The chain corresponding to the Tchebychev (or upper) mechanism is formed by links 1 (ground), 2, 3 and 4, as shown in Figure 3 so the closure equation for this case will be:

\[
r_1 e^{i\theta_1} + r_2 e^{i\theta_2} + r_3 e^{i\theta_3} + r_4 e^{i\theta_4} = 0
\]  

The Figure 3b shows the pantograph (or lower) mechanism is formed by links 5, 6, 7 and 8 and the closure equation is:

\[
\overrightarrow{MB} + r_6 e^{i\theta_6} + r_8 e^{i\theta_8} = 0
\]  

\( \overrightarrow{MB} \) is the vector joining points “M” and “B”. The position of point “B” is obtained from the solution of the first closure equation, whereas the position of point “M” is fixed in the hip or given by the movement of the linear actuators.

Finally, the stabilization or parallel mechanism is formed by links 9, 10, 11, 12 and the stabilization bar, see Figure 3. Here, we find two closed loops: the upper with links 6 (length from “B” to “C”), 9, 11 and the stabilization bar; and the lower with links 8 (length from “C” to “A”), 10, 11 and 12. So the two closure equations are the following:

\[
r_6 e^{i\theta_6} + r_7 e^{i\theta_7} = r_9 e^{i\theta_9} + r_{10} e^{i\theta_{10}}
\]  

\[
r_7 e^{i\theta_7} + r_8 e^{i\theta_8} = r_9 e^{i\theta_9} + r_{10} e^{i\theta_{10}}
\]  

Where \( \delta \) is the angle defined between the horizontal axis and the stabilization bar, counterclockwise.

Solving the implicit equation systems set out, it is possible to find the angular position of every link as a function of the input crank \( \theta_2 \). The solution of the equation system corresponding to the Tchebychev mechanism gives us the angles \( \theta_4 \) and \( \theta_3 \).

\[
\theta_4 = \cos^{-1}\left(\frac{2A_{aux} C_{aux} + 2B_{aux} \sqrt{B_{aux}^2 - C_{aux}^2} + A_{aux}^2}}{2(A_{aux}^2 + B_{aux}^2)}\right)
\]  

\[
\theta_3 = 2\pi - \cos^{-1}\left(\frac{-B_{aux} - r_4 \cos \theta_4}{r_3}\right)
\]  

Where \( A_{aux}, B_{aux} \) and \( C_{aux} \) are auxiliary variables dependant of \( \theta_2 \) defined to compact the resulting equation.
Solving the equation system corresponding to the closed loop of the pantograph mechanism, angles $\theta_8$ and $\theta_6$ can be obtained:

$$\theta_8 = -\cos^{-1}\left(\frac{-F_{aux}X_{MB} - Y_{MB} \sqrt{4r_8^2\left(X_{MB}^2 + Y_{MB}^2\right)} - F_{aux}^2}{2r_8\left(X_{MB}^2 + Y_{MB}^2\right)}\right)$$ \hspace{1cm} (13)

$$\theta_6 = \cos^{-1}\left(\frac{X_{MB} + r_6 \cos \theta_6}{r_6}\right)$$ \hspace{1cm} (14)

Where $F_{aux}$ is an auxiliary variable defined to compact the resulting equation.

The angle $\theta_7$ is found from a simple trigonometric relation between it and $\theta_6$, as shown in equation 15.

$$\theta_7 = \theta_6 - \pi = \cos^{-1}\left(\frac{X_{MB} + r_6 \cos \theta_6}{r_6}\right) - \pi$$ \hspace{1cm} (15)

Finally, solving the equations corresponding to the upper and lower sections of the stabilization mechanism, we obtain the expressions that determine the angles $\theta_9$ and $\theta_{11}$ first, and $\theta_{10}$ and $\theta_{12}$ then.

$$\theta_9 = -\cos^{-1}\left(\frac{a_gC_{aux} \pm b_g \sqrt{4r_9^2\left(a_g^2 + b_g^2\right) - C_{aux}^2}}{2r_9\left(a_g^2 + b_g^2\right)}\right)$$ \hspace{1cm} (16)

$$\theta_{11} = \cos^{-1}\left(\frac{r_{11} \cos \delta - r_c \cos \theta_7 + r_c \cos \theta_9}{r_{11}}\right)$$ \hspace{1cm} (17)

$$\theta_{10} = -\cos^{-1}\left(\frac{a_pC_{aux} \pm b_p \sqrt{4r_{10}^2\left(a_p^2 + b_p^2\right) - C_{aux}^2}}{2r_{10}\left(a_p^2 + b_p^2\right)}\right)$$ \hspace{1cm} (18)

$$\theta_{12} = \cos^{-1}\left(\frac{r_{12} \cos \theta_{11} - r_{11} \cos \theta_9 + r_{10} \cos \theta_{10}}{r_{12}}\right)$$ \hspace{1cm} (19)

Again, $a_g$, $b_g$, $a_p$, $b_p$, and $C_{aux}$ are auxiliary variables in order to reduce the size of the equations.

Once we know the angular position of every link it is trivial to know its position at any time, as well as the position of their centre of masses. Furthermore, taking the first time derivative of these equations will establish the angular velocities of every link (and the linear velocity of their centre of masses) and taking the second one, the angular accelerations (and the linear acceleration of their centre of masses). Defining the kinematics of one leg of the Mimbot robot.

As both legs are equal, but with an out of phase of 180 degrees, the whole kinematics of robot is defined.

### 4 A PARALLEL COMPUTING APPROACH

As stated in [18] following Flynn’s taxonomy of parallel computers [19], there are two major groups: in one hand, it is had the multiple instruction multiple data (MIMD) architectures, in the other hand, the single instruction multiple data (SIMD) architectures. The former refers to applications that execute multiple sequences of instructions of a program at the same time, while the latter execute the same sequence of instructions at the same time on multiple instances of data. The characteristics of GPU, with thousands of cores and their fast access to memory registers, combined with the nature of our problem, drive us to use the SIMD architecture to develop the parallelization of the mathematical model explained above.

In a typical PC, the memories of the CPU and GPU are physically distinct and separated by a PCI-Express bus. This is the way a programmer has to view things, data shared between the CPU and GPU must be allocated in both memories and copied between then in two the
directions. With CUDA 6 it is introduced one of the most dramatic programming model improvements in the history of this platform, Unified Memory (UM). This new feature creates a pool of managed memory that is shared between the CPU and GPU and accessible to both the CPU and GPU using a single pointer [20].

The key aspect, from a programming point of view, is that now the programmer only sees a pool of memory and doesn’t have to take care of allocating data in both memories and transfer data from one to another. UM dramatically simplifies memory management in GPU-accelerated applications at the time that allows a cleaner and more elegant code. Although UM doesn’t improve the speed of the execution, it makes the life’s programmer easier.

Before explaining the algorithm implementation carried out in this paper, it is a good practice to describe the original method developed to solve the mathematical model over MATLAB. As see in Figure 5, after the program starts, the link sizes, the input variables and the number of iterations and time increment for every step are set up. Next step is to choose the kinematic model to solve among the five predefined. After that, the 179 equations defining the model are computed for the first time step. The results obtained are stored in the memory and the index N counting the number of iteration, refreshed. If the last iteration has not achieved, the next time step will be computed, otherwise, the program exits the “for” loop. The obtained results

![Flowchart for the sequential algorithm](image)

**Figure 5.** Flowchart for the sequential algorithm

The key aspect, from a programming point of view, is that now the programmer only sees a pool of memory and doesn’t have to take care of allocating data in both memories and transfer data from one to another. UM dramatically simplifies memory management in GPU-accelerated applications at the time that allows a cleaner and more elegant code. Although UM doesn’t improve the speed of the execution, it makes the life’s programmer easier.
corresponding to the whole simulation are stored in several files and plotted for a visual analysis, ending the program here.

Given the problem characteristics and the hardware capabilities, a new parallel approach is proposed. This time, instead of a “for” loop to calculate every time step, we will have “n” number of threads, such as time steps to calculate. Obviously, these threads will be launched in parallel in the GPU, so each thread will compute each instant of time for all the output data of the subroutine executed.

Although MATLAB offers the chance of using CUDA functions, the equations obtained in section 2 have been implemented in CUDA C++. In this way, the programmer has full access to the memory and parallel threads management, so the code would be more efficient and the execution faster than that running over MATLAB; added to the fact that MATLAB is an interpreted language and C++, a compiled language.

The algorithm corresponding to the parallel approach to the problem is shown in Figure 6. The first steps are identical to the previous sequential version over MATLAB: after starting, the link sizes, the input variables and the number of iterations and time increment for every step are set up; as well as the kinematic model is chosen. According to this model, the set of equations to solve is arranged. At this point is where the differences between the two algorithms begin. First, the attributes of the available GPU are identified, that is, the maximum number of threads per
block \( (N_{\text{Mthread}}) \), of blocks \( (N_{\text{Mblock}}) \) and grid size \( (N_{\text{Mgrid}}) \). These numbers are very important because they determine how many threads it is possible to launch in parallel and how it must be done. After that the required Unified Memory to store all the variables that will be used by the GPU is allocated. Next, they are launched so many parallel threads as iterations have, which will compute the iterations. This process is made within the GPU capabilities, splitting the threads in the blocks needed to not exceed the maximum number of threads per block. The obtained results from the parallel computation are stored and memories of both GPU and CPU synchronized. Finally, as equal as done in the original algorithm, output data is stored in several files and results are plotted.

5 NUMERICAL RESULTS

We tested the GPU based parallel algorithm and compared it with the serial algorithm implemented over MATLAB in terms of computing time. For the results in Table 1, we carried out a total of ten groups of experiments (with 20 tests for group) were carried out using the two models of stabilizer and five different size problems, this is, the number of points to calculate. MATLAB is version R2013b running over a CPU Intel Xeon E5410 2.33GHz; the GPU is an NVIDIA GeForce GTX 660 Ti. The GPU algorithm is at least one order of magnitude faster than the serial algorithm.

The simulation time increases following a third grade polynomial with the problem size for the sequential version, Figure 7. In the case of the parallel code executed in the GPU, the simulation time increases almost linearly with the problem size, although, for unknown reasons, it exits a steady zone between 1000 and 5000 points, Figure 8.

<table>
<thead>
<tr>
<th>Size</th>
<th>Original stabilizer</th>
<th>New stabilizer</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MATLAB [ms]</td>
<td>CUDA C++ [ms]</td>
<td>Speed</td>
</tr>
<tr>
<td>201</td>
<td>49.94</td>
<td>0.58</td>
<td>85.82</td>
</tr>
<tr>
<td>601</td>
<td>142.30</td>
<td>0.92</td>
<td>155.04</td>
</tr>
<tr>
<td>1101</td>
<td>254.69</td>
<td>1.37</td>
<td>185.32</td>
</tr>
<tr>
<td>5001</td>
<td>1622.04</td>
<td>1.39</td>
<td>1163.34</td>
</tr>
<tr>
<td>10001</td>
<td>4619.67</td>
<td>2.69</td>
<td>1715.17</td>
</tr>
</tbody>
</table>

Table 1. Comparison of performance between CPU and GPU codes.

Figure 7. Execution time vs. problem size. This graph shows that the parallel algorithm scales almost linearly.
Figure 8. Execution time vs. problem size. This graph shows that the sequential algorithm scales following a polynomial law.

Figure 9 shows that, as expected, the performance of the GPU is better than the CPU as the problem size grows. Here the power of executing hundred or thousand threads in parallel is clearly visible, reaching speedups up to 1700 times. Unfortunately, it was not possible to go further in the comparison analysis due to the erratic behaviour of the sequential code when executing the algorithm with more than 10000 points.

5.1 Merit Indexes

In order to reach the best design it is needed to define a parameter (or a set of parameters) that evaluates correctly the goals we want to achieve. A lot of indexes can be considered for these tasks, according to the features under study. In this work, it will only be considered indexes based on the distance between trajectories

Hausdorff distance (Hd): The Hausdorff distance [21] measures how far two subsets of a metric space are from each other. The result is the distance between the farthest reciprocal points of both subsets or, in other words, the maximum of the largest minimum distance between the gaits compared.
Minimum distance: This distance measures how near two subsets of a metric space are from each other. It is the distance between the nearest reciprocal points of both subsets or, in other words, the minimum of the smaller minimum distance between the gaits compared.

Average distance: It is the average distance between the nearest reciprocal points of both subsets. It’s to say, the average distance between the trajectories of both gaits.

Although these indexes could be a good tool to determine the best design, they can offer erroneous results, especially if the gaits analysed intersect each other. To solve this issue, a new index is established. This parameter takes into account not only the distance between trajectories but also the similarity.

Scale factor amplitude (SFA): Given the reference and the simulated gaits, this index calculates the relation between the points of both sets. Once the gaits are centred in the origin of coordinates, the algorithm calculates the minimum distances between points. For each pair of nearest points it is calculated the distance from the origin to each of the two points and the relation between those distances, obtaining something like a scaling factor. After that, the amplitude of this scaling factor is computed. If the two gaits have similar shapes, the scaling factor should be constant, so the best solution will be the one whose amplitude is minimal. Figure 10 shows the evolution of the SFA index. As it can be observed, the minimum value is obtained for a link length of 540 mm, which corresponds to the most similar pair of gaits as seen in Figure 11.

Figure 10. Values of Scale Factor Amplitude (SFA) versus the variation of link dimension. The minimum (optimum) value is achieved for a length of 540 mm.

Figure 11. Comparison between the reference (dashed, inner, line) and the simulated gait. As observed, both gaits have the same shape.
Table 2 shows the optimal lengths obtained for a number of links using the scale factor amplitude. Data are divided in four groups according to the minimum and maximum lengths studied, as well as the minimum and maximum SFA reached. This way the outer limits are known, what gives us an idea on the SFA sensitivity to the variation of each link analysed. The optimal lengths correspond to the minimum SFA for every case studied and are remarked in bold.

<table>
<thead>
<tr>
<th>Link</th>
<th>Minimum length</th>
<th>Maximum length</th>
<th>Minimum SFA</th>
<th>Maximum SFA</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>L (mm)</td>
<td>SFA (L_{min})</td>
<td>L (mm)</td>
<td>SFA (L_{max})</td>
</tr>
<tr>
<td>rA</td>
<td>432</td>
<td>0.4685</td>
<td>647.5</td>
<td>0.5695</td>
</tr>
<tr>
<td>rB</td>
<td>120</td>
<td>0.5867</td>
<td>179.8</td>
<td>0.9402</td>
</tr>
<tr>
<td>rC</td>
<td>216</td>
<td>0.4886</td>
<td>288.6</td>
<td>1.905</td>
</tr>
</tbody>
</table>

6 CONCLUSIONS

This work presents the PASIBOT biped robot, a 1 DOF robot based on low cost technology, whose basic mechanical system consists of two legs resulting from the combination of classical mechanisms (Tchebychev mechanism, pantograph and a stabilization system). Its evolution into MIMBOT, a biped robot with 5 DOF, is outlined. It has also been developed a parametric mathematical model that describes in detail the kinematics of MIMBOT biped robot.

Two different approaches are presented to solve the kinematic problem, the first one using classical computation techniques over MATLAB and the second one taking advance of the latest advances in parallel computation over GPUs.

Tests carried out demonstrate the convenience of using parallel computation to solve the problem. The best results are obtained for the greater problem size, as in these cases the computation time required is three orders of magnitude lower. In addition, CUDA C++ allows full control over all the critical parameters involved in the process, as the memory management or the number and distribution of parallel threads launched.

Several tools have been proposed in order to quantify the similarity between simulated and reference gaits when the geometry of basic mechanisms of biped robot is modified. These tools have been applied to our data obtaining interesting results. The first three methods proposed only based in the distance between points yield incorrect solutions. However, the last tool based not only on the distance but also on the shape shows promising results. In fact, the SFA index allows us to obtain consistent results in the optimization process.

REFERENCES


Towards an Enhanced Controller for Slope Climbing

Athanasios S. Mastrogeorgiou*, Evangelos G. Papadopoulos#

* School of Mechanical Engineering
National Technical University of Athens
Heroon Polytechniou 9,
15780 Zografou, Greece
amast@mail.ntua.gr

# School of Mechanical Engineering
National Technical University of Athens
Heroon Polytechniou 9,
15780 Zografou, Greece
egpapado@central.ntua.gr

ABSTRACT
This paper presents simulation results obtained with a 3D model of the NTUA quadruped robot in the Webots simulation environment during slope climbing. Initially, the robot is controlled to perform pronking on level ground in order to validate the simulation environment. Gradually, the inclination is increased and simulations are conducted to discover the maximum value of positive slope the quadruped can cope with. Finally, disturbances are introduced and it is shown that the robot’s forward deceleration mainly depends on the front leg touchdown angles.

Keywords: quadruped, control, event driven, slope.

1 INTRODUCTION
In recent years, there has been an increasing interest in legged robots. A great deal of research for moving quadrupeds on uneven terrains, utilising observations of animals, has been carried out with significant results [1], [2]. Quadruped robot locomotion is a difficult task due to the high system complexity and the rough environment. In addition, critical stability issues emerge when considering multibody systems such as quadruped robots. Legged robots have complex dynamics and many degrees of freedom that must be well orchestrated for achieving a robust and dynamically stable locomotion pattern. Handling positive or negative slopes enhances the locomotion qualities of legged robots, but demands more from its actuation system. However, higher torque requirements have an adverse impact on a robot’s total mass, due to the need for bigger motors, which also require bigger batteries.

Legged robots have an advantage in dealing with various terrain types, or in handling terrain discontinuities with the use of accurate foot placement. Such systems have hybrid dynamics that are described by different sets of differential equations, according to the phase at which the robot is in (flight phase, double stance phase, etc.). Up to now, enhanced controllers, e.g. by means of computer vision [3], have been implemented for trotting on rough terrain. However, legged robots are difficult to control and as a result, they are subject to dangerous tipover instabilities.

Tipover prevention criteria have been introduced aiming at prevention of dangerous situations for mobile manipulator systems [4]. The most influential tipover stability measures are based on two criteria; the robot’s centre of mass (CM) and the support polygon defined by the convex area spanned between the ground contact points. The zero-moment point (ZMP) [5], originally derived for stabilizing bipedal robots, defines a point on the ground where the moment of total external forces (inertial plus gravity force) becomes zero. The Force-Angle algorithm measures stability by the angle of the total force applied to the centre of mass. The angles are referenced to the convex support polygon derived from the outer ground contact points of the robot. Building on this idea, Moosavian and Alipour proposed the Moment Height Stability (MHS) measure [6]. These criteria take into account tipover or rollover when operating over uneven terrain, and/or when exerting large forces or moments [7]. In past research, these tipover criteria have been validated in real-world scenarios [8].

In our current work, we investigate stability issues of quadruped robots on positive slopes subject to various disturbances. For this purpose, a 3D quadruped robot model has been
implemented in Webots 8 [9] and the equations of motion of a simpler passive 2D model have been calculated analytically. The Webots model is equipped with the necessary sensors (gyros, accelerometers, force sensors, laser range finder etc.) for state estimation and accurate phase triggering. The control algorithm uses sensor measurements to calculate the necessary torque and touchdown leg angles for stable pronking. Initially, the robot is controlled to perform pronking on level ground. Gradually, the inclination is increased. The quadruped’s performance is validated to be similar to [10]. We analyse how stable dynamic running can be performed as terrain morphology changes, how the quadruped’s gaits can be rearranged in order to carry out these tasks, and what makes a gait more persistent to disturbances compared to alternative ones.

In this work, we seek to enhance the controller in various ways by answering the previously stated questions. We examine the different support situations for quadrupeds. Firstly, during the double stance phase, the legs are in contact with the ground and form a support polygon. Experiments with the NTUA quadruped show that, during dynamic running, tipover may occur when the robot rotates around the (front) left toe – right toe axis (or back left and right toes respectively). In this case, we seek solutions in which the total force acting on the CoM is pointing towards the side of the robot with a leg about to contact the ground. Overall, no single approach for every terrain or inclination exists, but instead stable running also depends on friction constraints between the tow and the ground as well as on ground compliance.

Simulations will be performed for level ground and for a maximum slope of 20°. In the conducted experiments the quadruped model in Webots 8, performed pronking or bounding using a controller previously developed [11]. With this controller, forward velocity on lift-off and apex height, are maintained within desired limits, (Figure 4). In these simulations, by enhancing the controller with the use of events when a disturbance is sensed we show that the quadruped is able to cope with a sudden increase in the forward velocity and dangerous tipovers do not occur.

2 SIMMULATION SETUP

A 3D model of the NTUA quadruped was created in Webots 8 Simulation Environment, see Figure 1. The simulation setup can be seen in Figure 2.

![Figure 1](image)

Figure 1: The 3D model of the NTUA quadruped in Webots 8 Simulation Environment during dynamic running. The legs are compliant and the projection of the body’s CoM is within the robot’s support polygon. Pronking on a 20° slope has been achieved.

A control algorithm previously presented in [11] has been extended in order to be connected to the Robot Model in Webots. A Physics plugin for Webots 8 was created so that the controller could communicate with the Open Dynamics Engine [12]. During the flight phase, the Matlab controller calculated the touchdown angles for the front and rear legs and the torque to be applied by the hip servos during the next stance phase in order to maintain the desired apex height and forward velocity.
2.1 Description of the controller

The small footprint and the control on the apex height are important for traversing rough terrain since overcoming sudden obstacles such as rocks or handle discontinuities requires the foot to maintain a specific clearance from the ground. Thus, the MP controller [13] tries to maintain a passive gait with desired characteristics by applying proper actuation to compensate for energy losses. This approach results in minimum energy consumption, but requires an estimation of the leg compliance and system losses.

On liftoff the controller calculates a touchdown angle to achieve the desired apex height, and applies a constant torque during stance to achieve the desired forward velocity. As a result, the touchdown angle $\gamma_{td}$ to achieve a desired apex height is calculated as a function of the robot’s state at lift off [14]:

$$\gamma_{td} = f_1 \left[ m, k, L, d, I, b, g, \dot{x}_{des}, h_{des} \right]$$

(1)

where $\dot{x}_{des}$ and $h_{des}$ are respectively the desired forward velocity and apex height. During the flight phase, a simple proportional derivative (PD) controller is used to position rear (b) and front (f) legs at the desired touchdown angles. The control torque applied by the actuator is then,

$$\tau_{bf} = k_p \left( \gamma_{td,bf} - \gamma_{bf} \right) - k_d \dot{\gamma}_{bf}$$

(2)

where $k_p$ and $k_d$ are controller gains. The values of $k_p$ and $k_d$ are selected in the Webots simulation in order for the controller to be both fast enough to change the leg angle during the flight phase, while avoiding overshooting and unwanted oscillations. The necessary control torque ($\tau_s$) that must be applied during the stance phase, is also calculated at lift off as a function of the robot’s state:

$$\tau_s = f_2 \left[ m, k, L, d, I, b, g, \dot{x}_{des}, h_{des}, \gamma_{td} \right]$$

(3)

2.2 The Webots 3D model

The Webots 8 simulation environment was used to create a detailed model of the NTUA quadruped (Figure 3 & 6). The model consists of the main body, containing the electronics and the sensors needed for the controller, and four identical legs consisting of five individual parts that can be seen in Figure 3. The body can rotate at an angle ($\theta$) around the z-axis of its CoM. The rotation servo allows positioning of the legs at angle ($\gamma$) at the sagittal plane. Each leg has a passive prismatic joint modelled as a linear compression spring of constant ($k$) and viscous dumping coefficient ($b$). The prismatic joint allows changes of the front or rear legs length ($l_{bf/f}$) and energy accumulation during locomotion. Table 1 summarizes model and motion...
parameters. It should be noted that front and rear legs are modelled to have the same uncompressed length \(l_0\), spring constant \(k\) and viscous damping coefficient \(b\). The model parameters have been selected such as to obtain similar results to [10].

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
<th>Webots / Real model Value</th>
<th>2D model Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>((x, y, z))</td>
<td>CoM coordinates</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>(\theta)</td>
<td>body pitch angle</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>(b)</td>
<td>as index: back leg</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>(f)</td>
<td>as index: front leg</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>(\gamma_b, \gamma_f)</td>
<td>leg absolute angle</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>(\varphi_f, \varphi_b)</td>
<td>leg relative angle</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>(l_f, l_b)</td>
<td>current leg length</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>(x_{body}, y_{body}, z_{body})</td>
<td>robot body dimensions</td>
<td>0.6 m · 0.8 m · 0.35 m</td>
<td>0.6 m · 0.8 m</td>
</tr>
<tr>
<td>(K_s, K_f)</td>
<td>leg spring stiffness</td>
<td>3400 N/m</td>
<td>6800 N/m</td>
</tr>
<tr>
<td>(b)</td>
<td>viscous friction coefficient</td>
<td>8 N \cdot s/m</td>
<td>16 N \cdot s/m</td>
</tr>
<tr>
<td>(d)</td>
<td>hip joint to CoM distance</td>
<td>0.27 m</td>
<td>0.27 m</td>
</tr>
<tr>
<td>(l_0)</td>
<td>leg rest length</td>
<td>0.3 m</td>
<td>0.3 m</td>
</tr>
<tr>
<td>(l_{ul})</td>
<td>upper leg length</td>
<td>0.075 m</td>
<td>-</td>
</tr>
<tr>
<td>(l_{ll})</td>
<td>lower leg length</td>
<td>0.15 m</td>
<td>-</td>
</tr>
<tr>
<td>(m)</td>
<td>body mass</td>
<td>10.5 kg</td>
<td>10.91 kg</td>
</tr>
<tr>
<td>(J)</td>
<td>body inertia w.r.t. z-axis</td>
<td>0.8 kg \cdot m^2</td>
<td>0.8 kg \cdot m^2</td>
</tr>
<tr>
<td>(J_i)</td>
<td>leg inertia w.r.t. z-axis</td>
<td>0.0019 kg \cdot m^2</td>
<td>-</td>
</tr>
<tr>
<td>(l_m)</td>
<td>leg mass</td>
<td>0.41/4 kg</td>
<td>-</td>
</tr>
<tr>
<td>(m_{total})</td>
<td>total robot mass</td>
<td>10.91 kg</td>
<td>10.91 kg</td>
</tr>
</tbody>
</table>

Figure 3: The 3D model of the NTUA quadruped in Webots 8 Simulation Environment. The 4 legs are identical and consist of five individual parts (the rotational servo, the upper leg, the spring, the lower leg and the foot).

2.3 The Analytical 2D model

To obtain a better insight into the dynamics of the quadruped, the equations of motion of a 2D lumped parameter model of a quadruped were derived with a Lagrangian approach, using body Cartesian coordinates, \(x\), \(y\), and pitch, \(\theta\). The model consists of two compliant virtual legs (VLegs) respectively. The Vlegs are indicated as rear (b) or front (f). A VLeg, front or rear,
models the two respective physical legs that operate in pairs when a gait is realized and exerts equal torques and forces on the body as the set of the physical ones [15]. The planar model is valid for gaits that have symmetry about the plane of the forward motion, like pronking and bounding.

The planar quadruped robot model can be seen in Figure 5. It should be noted that front and rear legs are modelled to be massless linear springs. The 2D analytical lumped parameter model of a quadruped is shown in Figure 5.

![Figure 4: Planar quadruped robot model.](image)

The equations of motion for the double stance and flight phase were developed and are presented here. Front and back stance equations of motion can be derived from the double stance ones, by cancelling the terms that do not appear in the front or back stance respectively. All symbols in the equations are described in Table 1.

### Double Stance

\[ m\ddot{x} + K_f (L - l_f) \sin \gamma_f - K_f (L - l_f) \sin \gamma_f = 0 \]  \hspace{1cm} (4)

\[ m\ddot{y} - K_b (L - l_b) \cos \gamma_b - K_f (L - l_f) \cos \gamma_f + mg = 0 \]  \hspace{1cm} (5)

\[ I\ddot{\theta} + dK_b (L - l_b) \cos (\gamma_b - \theta) - dK_f (L - l_f) \cos (\gamma_f - \theta) = 0 \]  \hspace{1cm} (6)

### Flight Phase

\[ \ddot{x} = 0, \ddot{y} = 0, \ddot{\theta} = 0 \]  \hspace{1cm} (7)

### Front Stance

\[ m\ddot{x} + K_f (L - l_f) \sin \gamma_f = 0 \]  \hspace{1cm} (8)

\[ m\ddot{y} - K_f (L - l_f) \cos \gamma_f + mg = 0 \]  \hspace{1cm} (9)

\[ I\ddot{\theta} - dK_f (L - l_f) \cos (\gamma_f - \theta) = 0 \]  \hspace{1cm} (10)

### Back Stance

\[ m\ddot{x} + K_b (L - l_b) \sin \gamma_b = 0 \]  \hspace{1cm} (11)

\[ m\ddot{y} - K_b (L - l_b) \cos \gamma_b + mg = 0 \]  \hspace{1cm} (12)

\[ I\ddot{\theta} + dK_b (L - l_b) \cos (\gamma_b - \theta) = 0 \]  \hspace{1cm} (13)
To evaluate the model, initial experiments have been carried out on level ground. As it can be seen in the phase plane diagram in Figure 5 and the forward velocity in Figure 7 the quadruped can perform stable dynamic running.

![Phase plane diagram of the vertical position and acceleration of the robot’s CoM while performing dynamic running on level ground.](image)

**Figure 5:** Phase plane diagram of the vertical position and acceleration of the robot’s CoM while performing dynamic running on level ground.

The Webots model of the robot in comparison to the real robot is shown in Figure 5.

![Constrast of the NTUA quadruped and the simulated robot model in Webots 8.](image)

**Figure 6:** Constrast of the NTUA quadruped and the simulated robot model in Webots 8.

### 3 RESULTS

The controller can achieve stable dynamic running on inclinations ranging from 0 to 20 degrees. In Figure 7 simulation results for 10 degrees slope climbing are presented. The quadruped can perform repeatable motions (steady state) in all these cases.
The controller has been also tested on various disturbances while the quadruped performs dynamic running. In Figure 8, a horizontal force of 1200 N (in the same direction of the robot’s motion) is applied at the CoM of the body for 4ms tending to increase the forward velocity instantly. The controller overcomes this disturbance and the system reaches again a steady state.

If the disturbance applied is two times greater than the previous case, the controller fails to retain the robot in its standing position. As the forward velocity increases instantly when the front feet contact the ground, the pitch angle also starts to increase due to the frictional forces. Eventually, the robot falls since the front legs cannot prevent it.

In Figure 9, the body pitch angle over time is depicted. This time the robot falls because of the applied disturbance. An enhancement of the controller (which can also be extended for 3D motions) in order to overcome such disturbances is developed. Since the controller is robust enough to handle disturbances such as those in Figure 8, an event that is triggered for higher values of the forward velocity is introduced.
Figure 9: Pitch angle over time. The disturbance is applied at time $t=2363$ ms.

Figure 10: The robot remains in standing position despite the disturbance. When a rapid change in the forward velocity (in this case from 1m/s to 2.7m/s) is sensed the controller reacts by increasing the front legs touchdown angle (in this case from 0.19 rad to 0.39 rad).

The front leg hip joint is able to move from -0.43 rad to 0.43 rad. At the first stage, experiments using the maximum angle that the front hip joint can reach (0.43 rad) were carried out. As a result, the robot was able to overcome the applied disturbance as shown in Figure 10. Because of the fact that, the quadruped reaches a steady state, the duration of the flight phase and stance phase can be estimated. The stance phase lasts for about 0.1s. As a result, the disturbance has to be cancelled out, or as shown in Figure 8, reach a value that the controller can handle, in less than 0.1s. Using the equation of motion of the simplified 2D model for the front stance phase, it can be seen that the forward acceleration of the model depends on the touchdown angle of the robot’s front legs ($\gamma$).
The touchdown angle of the front legs in the steady state (for 1 m/s desired forward velocity) is \( \approx 0.19 \) rad. The introduced disturbance increases the forward velocity of the quadruped from the desired 1 m/s to 2.3 m/s. In Figure 8 it is shown that for minor disturbances there is no need to take action while for situation such as the one in Figure 9, it is needed. According to the equations of motion for the front stance \( \ddot{x} \) depends on \( \sin \gamma_f \) ranging from \( \sin -0.43 = -0.417 \) to \( \sin 0.43 = 0.417 \) and \( (L - l_f) \) ranging from 0 to 3.5 cm. If it is needed to increase the forward deceleration because of a disturbance the term, \( (L - l_f) \), has a minor impact in that. According to Figure 11, the spring compression, \( (L - l_f) \), in two different cases is similar. As a result, experiments have shown that by increasing the touchdown angle of the front legs from 0.19 rad to 0.29 rad (from \( \sin 0.19 = 0.189 \) to \( \sin 0.29 = 0.286 \) is enough to decelerate the robot from 2.3 m/s to 1.5 m/s and retain the quadruped in standing position. In addition, by increasing the front leg touchdown angle by 0.2 rad, the quadruped can handle disturbances 4 times the one shown in Figure 8 (from 2.7 m/s to 1.5 m/s).
Figure 12: The robot remains in standing position despite the disturbance. When a rapid change in the forward velocity (in this case from 1 m/s to 2.3 m/s) is sensed the controller reacts by increasing the front legs touchdown angle (in this case from 0.19 rad to 0.29 rad).

In conclusion, since \( \sim 0.19 \) rad is the touchdown angle of the front hip joints in the steady state and the controller is robust to handle disturbances up to 0.5 m/s higher than the desired forward velocity (1 m/s) we can use the addition 0.23 rad (0.43 rad is the mechanical limit of the joint) to enhance stability.

4 CONCLUSIONS

This paper presented simulation results obtained with a simulated quadruped robot on sloping ground in Webots. It has been shown that in situation such as those presented, events can be introduced and dangerous tipover subject to disturbances can be prevented. In the first stage, experiments were conducted in order to find out how robust the controller is. In the second stage, an event was introduced. The whole range of the robot’s joint was used in order to keep the robot in standing position. In the end, this approach was enhanced since from our analysis it is shown that despite the various disturbances the spring compression and decompression remains almost the same. As a result, we can benefit only from the touchdown angle of the front legs in order to decelerate the robot. Finally, intermediate steps where introduced according to the intensity of the disturbance.

5 ACKNOWLEDGEMENTS

This research has been co-funded by the European Union (European Social Fund –ESF) and Greek national funds through the Operational Program “Education and Lifelong Learning” of the National Strategic Reference Framework (NSRF) – Research Funding Program: Thalis. Investing in knowledge society through the European Social Fund.

6 REFERENCES


Using the Generalized Inverted Pendulum to generate less energy-consuming trajectories for humanoid walking

Sahab Omran∗, Sophie Sakka#, Yannick Aoustin∗

∗ L’UNAM, IRCCyN, CNRS, Ecole Centrale de Nantes, University of Nantes, 1, rue de la Noë BP 92101. 44321 Nantes, France
[sahab.omran, yannick.aoustin]@irccyn.ec-nantes.fr

# IRCCyN, University of Poitiers, France.
sophie.sakka@irccyn.ec-nantes.fr

ABSTRACT
This paper proposes an analysis of the effect of vertical position of the pivot point of the inverted pendulum during humanoid walking. We introduce a new feature of the inverted pendulum by taking a pivot point under the ground level allowing a natural trajectory for the center of pressure (CoP), like in human walking. The influence of the vertical position of the pivot point on energy consumption is analysed here. The evaluation of the energy consumption is done using a sthenic criterion in a 3D dynamics simulation of the humanoid robot ROMEO (Aldebaran Robotics) and shows a consequent reduction of the robot torque solicitation with a pivot point under the ground.

Keywords: Inverted pendulum, pivot point, walking robots, energy consumption, joint torques.

1 INTRODUCTION
The linear inverted pendulum (LIP) was proposed by Kajita et al. to generate humanoid walking trajectories [4, 1]. This model was widely applied to various bipedal robots like HRP-2 [8], Asimo [9], and UT-Theta [10]. The main advantage of this approach is the simplicity of the dynamics and the analytical solution. The Generalized Inverted Pendulum model (GIP) [6] is a particular case of inverted pendulum models with a pivot point under the ground level in opposition to all other inverted pendulum models (for robot control or human motion) with a pivot point at ground level. It describes human normal walking from the external forces point of view, taking into account the mechanism of foot.

This work uses the GIP model [6], initially proposed to model human walking, to generate a more human inspired walking pattern of humanoid robots. The vertical component of the pivot point has been determined in a manner to minimize the energy consumed by the system.

This paper is organised as follows: First, we show that the dynamics equations system does not change if the pivot point is located under the ground or at ground level. Then, we show the effect of the vertical position of the pivot point on the inverted pendulum energy. After, we present a simulation results on the humanoid robot Romeo using trajectories with different depth of the pivot point.

2 Dynamics equation for an inverted pendulum with a pivot point under the ground
In this section, we will study the dynamics of the inverted pendulum in the general case: The CoM is not constrained to maintain a constant height during the movement and whose pivot point is located under the ground. Fig. 1 illustrates the inverted pendulum in the sagittal plane, where the mass $M$ moves under the force $f$ and the gravity $g$. The mass is connected to the ground with a massless rod. The pivot point of the pendulum is a virtual revolt joint located under the ground, $z_p$ is its vertical component. $\theta$ is the angle between the rod and the vertical axis. The global coordinates system is defined by the forward axis $\vec{x}$, the up-ward vertical axis $\vec{z}$ and the transversal axis $\vec{y} = \vec{z} \times \vec{x}$.

The force $f$ and the gravity create an acceleration of the CoM $[\ddot{x} \ \ddot{y} \ \ddot{z}]^T$. Along the vertical axis $z$:

$$f_z - Mg = M \ddot{z}$$  (1)
\[ f \cos \theta - Mg = M \ddot{z} \]  
(2)

\[ f = \frac{M(g + \ddot{z})}{\cos \theta} \]  
(3)

Along the forward axis \( x \):

\[ f_x = M \ddot{x} \]  
(4)

\[ f \sin \theta = M \ddot{x} \]  
(5)

By substituting Eq. (3) in Eq. (5), we obtain

\[ M \left( g + \ddot{z} \right) \frac{\sin \theta}{\cos \theta} = M \ddot{x} \]  
(6)

\[ \frac{\sin \theta}{\cos \theta} = \tan \theta = \frac{x}{z - z_p} \]  
(7)

The minus sign of \( z_p \) is justified by the fact that \( z_p \leq 0 \).

\[ (g + \ddot{z}) \frac{x}{z - z_p} = \ddot{x} \]  
(8)

\[ x = \frac{z - z_p}{g + \ddot{z}} \ddot{x} \]  
(9)

In a similar manner, we obtain the motion equation in the frontal plane.

\[ y = \frac{z - z_p}{g + \ddot{z}} \ddot{y} \]  
(10)

The motion of the CoM is characterized by the second order differential equations (9) and (10). These equations are very similar to those in the case where the pivot point is located at ground level. The only difference is that the term \( z \) is replaced by \( z - z_p \). Before the term \( z_p \) was sometimes used to express the vertical component of origin of the frame attached to the foot with respect to the global coordinate system and the pivot point was always in the foot. But now, \( z_p \) expresses the vertical distance between the pivot point and the ground. The differential equations 9 and 10 can be solved analytically when \( z = \text{const} \). When \( z \neq \text{const} \), these two equations are solved numerically.
3 Modeling

Let us consider a humanoid robot composed of \( n_{\text{act}} \) actuators to control its body movements in 3D. Let \( \mathbf{q} \) and \( \mathbf{X} \) denote the system generalized and operational coordinates vectors, respectively. We introduce the following notations:

\[
\begin{align*}
\mathbf{X} & \quad (6 \times 1) \quad \text{Absolute position and orientation of the waist;} \\
\mathbf{X}_{fi} & \quad (6 \times 1) \quad \text{Absolute position and orientation of foot} \ i \ (i = 1, 2); \\
\mathbf{q}_{\text{legi}} & \quad (n_{\text{leg}} \times 1) \quad \text{Actuated joints of leg} \ i \ (\text{ankle, knee and hip joints}); \\
\mathbf{q} & \quad (n \times 1) \quad \text{Vector grouping the controlled variables} \ (n = n_{\text{act}} + 6).
\end{align*}
\]

The vector \( \mathbf{q} \) contains the joint variables and the 3D position and orientation of the frame \( R_0 \) fixed in the left foot. The reference frame is defined such that \( x_0 \) denotes the horizontal advancement direction, \( z_0 \) is the vertical bottom-up direction, and \( y_0 = z_0 \times x_0 \) is the lateral direction.

3.1 Walking cycle – Assumptions

The walking cycle is defined by two successive steps (right and left legs). One step is composed of a single support phase (SS) on the stance leg, delimited by swinging foot takeoff and swinging foot strike, and a double support phase (DS) where the body weight is distributed on both legs, delimited by swinging foot strike and other foot takeoff. In what follows, the following assumptions will be considered for the complete motion:

- **A1** There is no rotation of the swing foot and the basin of the biped with respect to the roll, pitch, and yaw axes.
- **A2** The stance foot has a flat contact on the ground;
- **A3** Feet velocity and acceleration are equal to zero at foot strike. Thus no impact is considered.
- **A4** The CoM and the waist segment have the same linear velocity and acceleration profiles;

As humanoid systems are highly redundant, these assumptions allow setting the control schemes while reducing the redundancy order by setting arbitrarily several parameters. The motion of the swinging foot is defined as a polynomial function, where the polynomial coefficients were determined using initial and final positions, velocities, and accelerations.

3.2 Kinematics

For given desired Cartesian trajectories of the waist \( \mathbf{X} \) and the feet \( \mathbf{X}_{fi} \ (i = 1, 2) \), the inverse kinematics model leads to the values of the desired joint variables as detailed in Eq. (11).

\[
\begin{bmatrix}
\mathbf{V}_{fi} \\
\omega_{fi}
\end{bmatrix} =
\begin{bmatrix}
\mathbf{I}_{3 \times 1} & -\mathbf{L} \\
\mathbf{0}_{3 \times 1} & \mathbf{I}_{3 \times 1}
\end{bmatrix}
\begin{bmatrix}
\mathbf{V} \\
\omega
\end{bmatrix}
+ \mathbf{J}_{\text{legi}} \dot{\mathbf{q}}_{\text{legi}} \quad (i = 1, 2)
\]

(11)

where \( \dot{\mathbf{X}}_{fi} = [\mathbf{V}^T \omega_{fi}^T]^T, \dot{\mathbf{X}} = [\mathbf{V}^T \omega^T]^T, \mathbf{J}_{\text{legi}} \) denotes the \( 6 \times n_{\text{leg}} \) Jacobian matrix associated to the \( i \)-th leg, \( \mathbf{L} \) is the position vector between the waist and foot \( f_i \) and \( ^\ast \) is the skewsymetric matrix. Due to assumption A1, \( \omega = \omega_{fi} = 0 \), thus Eq. (11) becomes

\[
\dot{\mathbf{q}}_{\text{legi}} = \mathbf{J}_{\text{legi}}^{-1} \begin{bmatrix}
\dot{\mathbf{X}}_{fi} - \dot{\mathbf{X}} \\
\mathbf{0}_{3 \times 1}
\end{bmatrix} \quad (i = 1, 2)
\]

(12)

The desired \( n \times 1 \) controlled velocities vector \( \dot{\mathbf{q}} \) can be rebuilt as follows.

\[
\dot{\mathbf{q}} = \left[ \dot{\mathbf{q}}_{\text{leg1}} \quad \dot{\mathbf{q}}_{\text{leg2}} \quad \dot{\mathbf{q}}_{\text{free}} \quad \dot{\mathbf{X}} \right]^T
\]

(13)

where \( \dot{\mathbf{q}}_{\text{free}} \) denotes joint velocities of trunk and arms which can be set freely.

The rank of the Jacobian matrix of each leg was verified at each sampling period of motion to ensure that there is no singularity.
3.3 Dynamics

The dynamics of the system may be described by the three following equations.

\[
\begin{aligned}
D\dddot{q} + C\dot{q} + G &= B\Gamma + J^\top_1 R_1 & \text{if in single support, leg 1} \\
D\dddot{q} + C\dot{q} + G &= B\Gamma + J^\top_2 R_2 & \text{if in single support, leg 2} \\
D\dddot{q} + C\dot{q} + G &= B\Gamma + J^\top_1 R_1 + J^\top_2 R_2 & \text{if in double support}
\end{aligned}
\] (14)

The matrices \(D(q), C(q, \dot{q})\) and \(G(q)\) describe respectively the inertia, Coriolis and gravity forces acting on the system. \(\Gamma\) is the vector of the actuator torques \(\Gamma_i, i = 1, \ldots, n_{\text{act}}\). The matrix \(B\) is the actuation matrix; it expresses the contribution of each joint torque in the virtual work \(\delta w\):

\[
\delta w = \Gamma_1 \delta q_1 + \Gamma_2 \delta q_2 + \ldots + \Gamma_{n_{\text{act}}} \delta q_{n_{\text{act}}} = \delta q^\top B\Gamma
\]

where \(\Gamma = [\Gamma_1 \Gamma_2 \ldots \Gamma_{n_{\text{act}}}]^\top\) and \(B = [0_{n_{\text{act}} \times 6} \ 1_{n_{\text{act}}}]^\top\). The vectors \(R_1\) and \(R_2\) are the ground reaction forces exerted on foot 1 and foot 2 respectively.

In single support, there are \(n\) unknown variables which are the components of \((\Gamma, R_1)\) or \((\Gamma, R_2)\) depending on which foot is in contact with the ground. So, the \(n\) independent equations in the two first lines of Eq. (14) are sufficient for solving. On the other hand, in double support there are \(n + 6\) unknown variables in \((\Gamma, R_1, R_2)\) and only \(n\) equations available. In order to solve the problem in double support, six variables should be chosen and set to completely describe the system dynamics. The variables we choose are the six components of the ground reaction forces exerted on the foot that was supporting before the considered double support. Similarly to Omran et al. [5], these components are defined as third-order polynomial functions of time ensuring the continuity of the ground reaction forces with the two single support phases around the considered double support.

4 Energy consumption

Many criteria exist to evaluate energy consumption of a mechanical system, however to our best knowledge there is no ideal criterion [3]. In this approach, we chose the sthenic criterion which is defined by the integral of the quadratic actuators torques per unit of distance, as shown in Eq. (15). Its physical meaning is to be an image of the Joule effects if the actuators are DC motors. Furthermore, the torque amplitudes are decreased with the minimization of this criterion [2]. Then if we design an optimal walking gait with this criterion we can limit the weight of the needed motors.

\[
C_t = \frac{1}{d} \int_{t_0}^{t_f} \Gamma^\top \Gamma dt
\]

where \(t_0\) and \(t_f\) denote the beginning and ending instants of the total observed motion, \(d\) is the travelled distance. The sthenic criterion is a quantity proportional to the energy solicitation by actuators per unit of distance, while the quadratic torque deals with the instantaneous norm of motor torques: \(E_0(t) = \Gamma^\top \Gamma\).

5 Simulation settings

For the validation of method, we use the model of the 33 degrees-of-freedom (dof) humanoid robot ROMEO[7]. The robot total weight is 40.53 kg and its height is 1.43 m. ROMEO dof are distributed as follows: 6 per leg, 1 for each toe, 7 per arm, 1 for the trunk, 2 for the neck and 2 for the head. As we focus on the locomotion, only the 12 dof of the legs were controlled in motion and the other dof were set to zero. The kinematic chain of the lower body of Romeo is shown in Fig. 2. To show the effect of the vertical position of pivot point, the robot performs one step.
forward on a flat surface. The step length is set to 0.4 m, the step width is set to 0.192 m and the step duration is 0.5 s. The height of the CoM is constant \( z_c = 0.64 \) m. The horizontal position of the pivot point is set in the left foot center, so the robot takes one step forward with its right foot, as shown in Fig. 3. This motion is done many times, with different depth of the pivot point \( z_p \) taken in the range \([0, -1]\) m. After calculating the CoM trajectory in the horizontal plane, we obtain joint angles using the inverse kinematic model, then we calculate the joint torques using the dynamic model. In the end, we calculate the sthenic criterion for each trajectory.

6 Results
The resulting trajectories for each \( z_p \) are compared in terms of dynamic balance, joint torques and energy consumption.
6.1 Horizontal trajectory of CoM

The motion described in Sec. 5 is realized by the robot Romeo in five cases corresponding to five values of \( z_p \): \([0, -0.25, -0.5, -0.75, -1]\) m. Fig. 4 shows the fives resulting CoM trajectories in \( x \) and \( y \) directions as a function of time. We note that the \( x \) component of the CoM comes closer to a straight line when \(|z_p|\) increases. On the other hand, the oscillation amplitude of the \( y \) component decreases when the pivot point goes farther under the ground.

![Figure 4: Horizontal trajectory of CoM as a function of time for five values of \( z_p \)](image)

6.2 Trajectory of CoP

Fig. 5 represents two inverted pendulums. The first one with a pivot point at ground level (\( z_p = 0 \)), its motion in the sagittal and frontal planes is represented in Fig. 5(a) and Fig. 5(c) respectively. The second one with a pivot point under ground level (\( z_p = -1 \) m), its motion in the sagittal and frontal planes is represented in Fig. 5(b) and Fig. 5(d) respectively.

In Fig. 5, the CoM is represented by circles and the CoP is represented by triangles. The ground level (\( z = 0 \)) is represented by a green line. We notice that the CoP coincides with the pivot point when \( z_p = 0 \). In this case, the CoP is a fixed point. But when the pivot point is under the ground level, the CoP moves in \( x \) and \( y \) directions as we can see in Fig. 5(b) and Fig. 5(d) respectively. For this reason, we must verify that the distances travelled by the CoP in \( x \) and \( y \) directions are smaller than the robot foot dimensions before applying the trajectory to a humanoid robot.

To describe the relation between the CoP trajectory and the depth of the pivot point, we consider three inverted pendulums having the same parameters of step length \( L \), step width \( L_w \), and the CoM height \( z_c \). The pivot point depth for these pendulums are \( z_{p1} = 0 \), \( z_{p2} \) and \( z_{p3} \) such as \( 0 < |z_{p2}| < |z_{p3}| \). These three pendulums are illustrated in Fig. 6 in the sagittal and frontal planes. The foot is also represented in this figure by a bold green line.

We notice that the travelled distance by the CoP in \( x \) direction, increases when the pivot point depth increases (\( L_2 < L_3 \)). Along the \( y \) direction, the CoP does not move when the pivot point is at ground level, but it moves in a range when the pivot point is under the ground. The range of the CoP along the \( y \) axis is increasingly far from the foot center when the \(|z_p|\) increases. As we can see that \( l_2 \) is closer to the foot center than \( l_3 \).

For the bigger values of \(|z_p|\), the CoP trajectory may leave the foot. For example, in Fig. 6, the CoP of the pendulum corresponding to \( z_{p3} \) leaves the contact zone between the foot and the ground. Therefore, for each foot size and step length and width, there is an upper limit of \(|z_p|\) that keeps the CoP inside the foot.

For a robot with feet of length \( L_x \) and width \( L_y \), performing a step of length \( L \) and width \( L_w \), the
Figure 5: The inverted pendulum motion in sagittal and frontal planes in two cases: \( z_p = 0 \) and \( z_p = -1 \) m

Figure 6: The inverted pendulum during one step for three different pivot points.

upper limit of \(|z_p|\) respects the following relation:

In the sagittal plane:

\[
\frac{2L_x}{L} = \frac{|z_{p,\text{max}}|}{|z_{p,\text{max}}| + z_0} \quad \text{if} \quad L > L_x
\]
In the frontal plane:

\[
\frac{L_y}{L_w} = \frac{|z_{p,\text{max}}|}{|z_{p,\text{max}}| + z_0} \quad \text{if} \quad L_w > L_y
\]  

(17)

From these two equations, we can deduce the limit of \( z_p \):

\[
|z_{p,\text{max},\text{sagittal}}| = \frac{2 L_x z_0}{L - 2 L_x} \quad \text{and} \quad |z_{p,\text{max},\text{frontal}}| = \frac{L_y z_0}{L_w - L_y}
\]  

(18)

The upper limit of \( |z_p| \) is chosen as:

\[
|z_{p,\text{max}}| = \min(|z_{p,\text{max},\text{sagittal}}|, |z_{p,\text{max},\text{frontal}}|)
\]  

(19)

When we generate trajectories for experiments on a real robot, we should consider a security margin for CoP before calculating \( |z_{p,\text{max}}| \). The security margin can be defined as a percentage of foot dimensions. In simulation, the security margin may be not considered.

The humanoid robot Romeo feet are 0.289 m in length and 0.121 m in width. By applying Eq. (19), we obtain: \( |z_{p,\text{max}}| = 1.09 \text{ m} \). If we consider a margin of security of 50% of the foot, we obtain \( |z_{p,\text{max}}| = 0.29 \text{ m} \). The robot foot with the security margin is illustrated in Fig. 7.

![Figure 7: The foot and the CoP area.](image)

6.3 Joint torque

In this section, we consider two trajectories for humanoid robot Romeo. The first one was calculated using an inverted pendulum with a pivot point at ground level (\( z_p = 0 \)) and the second one was calculated using an inverted pendulum with a pivot point under the ground (\( z_p = -z_{p,\text{max}} = -0.29 \text{ m} \)). The two inverted pendulums have the same parameters for step length, step width, step duration and CoM height as given in Sec. 5. We compare joint torques for these two trajectories.

Fig. 8 shows the torques at the legs joints. The support leg contains six joints: ankle roll (\( \Gamma_2 \)), ankle pitch (\( \Gamma_3 \)), knee pitch (\( \Gamma_4 \)), hip pitch (\( \Gamma_5 \)), hip roll (\( \Gamma_6 \)), and hip yaw (\( \Gamma_7 \)). The swing leg contains six joints also: ankle roll (\( \Gamma_{13} \)), ankle pitch (\( \Gamma_{12} \)), knee pitch (\( \Gamma_{11} \)), hip pitch (\( \Gamma_{10} \)), hip roll (\( \Gamma_9 \)), and hip yaw (\( \Gamma_8 \)). In global, the two trajectories show similar behaviour. We can see that \( \Gamma_5, \Gamma_{10}, \Gamma_{11}, \) and \( \Gamma_{12} \) are almost the same. For these joints, the torque is a little reduced (between 1.8% and 5.2%) with a pivot point under the ground than with a pivot point at ground level.

For \( \Gamma_3, \Gamma_4, \Gamma_7, \Gamma_8, \) and \( \Gamma_9 \), we can observe two or three peaks for each torque. The torque amplitude peak-to-peak is reduced for \( z_p = -z_{p,\text{max}} \) compared to the case \( z_p = 0 \) by 20%, 10.8%, 1.3%, 20.2% and 26% respectively.

We can notice that main differences happen at ankle roll for the two legs \( \Gamma_2 \) and \( \Gamma_{13} \). \( \Gamma_2 \) is reduced when \( z_p = -0.29 \text{ m} \) compared to \( z_p = 0 \), we give the reduction rate at three moment within the
Figure 8: Joint torques for legs joints of Romeo for 2 values of the pivot point depth.

motion: 9.8% at (t=0 s), 91% at the peak (t=0.31 s) and 58.8% at the end of the step (t=0.5 s). In the same manner, $\Gamma_{13}$ is also reduced when $z_p = -0.29$ m compared to $z_p = 0$, we give the reduction rate at the three peaks: 29.4% at (t=0.1 s), 21.6% at (t=0.25 s) and 29.5% at (t=0.4 s).

The torque at the hip roll of the support leg $\Gamma_6$ is the only one that increases when the pivot point is under the ground level. The rate of increase with respect to the case $z_p = 0$ is given at three moment within the motion: 6.22% at (t=0 s), 1.97% at (t=0.31 s) and 6.39% at (t=0.5 s).
6.4 Energy evaluation

The profile of the quadratic torque $E_0 = \Gamma^\top \Gamma$ during one step is shown in Fig. 9. The simulations were run with two values of the pivot point depth: $z_p = 0$ and $z_p = -0.29$ m. The two graphs of

![Figure 9: Quadratic torque](image)

Fig. 9 show a peak of value at the beginning of single supports followed by a valley at midstance. Fig. 9 shows that $E_0$ values at the beginning and at the end of the single support are lower with a pivot point under the ground level that with a pivot point at ground level. The evolution of

![Figure 10: Sthenic criterion](image)

the sthenic criterion as a function of the pivot point depth is shown in Fig. 10, for values of $z_p$ between $[0 - 1.09]$ m, which is the maximum authorized in simulation. The situation $z_p = -0.718$ m minimizes the sthenic criterion, with the criterion value being reduced by 10.5% in comparison to the case $z_p = 0$. When we consider a security margin of 50% of foot dimensions, $z_{p,\text{max}} = -0.29$ m corresponds to 9% reduction of the sthenic criterion compared to the case $z_p = 0$. 

1536
7 CONCLUSIONS

This paper proposed an analysis of the effect of the vertical position of the pivot point on the energy consumption for humanoid walking gait. A 3D simulation was proposed to compare the classical inverted pendulum with a pivot point on the ground level and an inverted pendulums with a pivot point under the ground level. The dynamics analysis showed that the use of a pivot point under the ground reduced considerably the torque solicitation especially in the beginning of the single support. Moreover, the sthenic energy can be minimized for an optimal pivot point depth. The results can be included in walking pattern generators in order to reduce energy consumption during walking and to obtain a natural rolling of feet.

REFERENCES


A framework for data exchange and benchmarking of frictional contact solvers in multibody dynamics

Claude Lacoursière*, Mattias Linde†, Ying Lu#, Jeff Trinkle#

* HPC2N/UMIT Research Lab † Dept. of Computing Science
Umeå University ‡ Dept. of Computer Science
901 87 Umeå, Sweden 110 8th Street
claude@hpc2n.umu.se linde@cs.umu.se
# Dept. of Computer Science
Rensselaer Polytechnic Institute
110 8th Street
Troy, NY 12180, U.S.A.
rosebudflyaway@gmail.com trink@cs.rpi.edu

ABSTRACT

We present an HDF5 layout specification to store and exchange the run-time kinematic data of simulations of contacting multibody systems subject to dry friction in descriptor form. This is intended to be used to test solvers on the incremental problem defined by one single configuration, allowing any mathematical formulation and friction law, and compatible with any numerical method. We also introduce metrics to measure the quality of a solution comprehensively, separating the errors from the different aspects of a given friction law. We started a public web site with a collection of problems, from simple stacking to full vehicle dynamics as well as grasping robots. We provide software to manipulate the data contained in the HDF5 files, assemble matrices needed for different formulations, and to interface with existing solvers. We also provide software to compute our metrics and generate plots which provide a global picture of performance and accuracy. As this is written in MATLAB and OCTAVE we believe our contributions will allow anyone wanting to write or test a solver to work with “real life” examples without any difficulty, and concentrate on the numerical methods themselves, and get results before writing a code suitable for integration with a full featured software package.

Keywords: multibody dynamics, benchmark, solvers, friction, complementarity, contact mechanics, data representation.

1 INTRODUCTION

There is currently no entirely satisfactory solver for the computation of contact forces for multibody systems subject to dry friction laws. Issues include performance, stability, robustness, and accuracy. There are also multiple physical models based either on points or contacts for instance, multiple formulations of the mathematical problem [2] as Nonlinear or Linear Complementarity Problems, Variational Inequalities [10], and optimization problems [9]. There are then numerous numerical methods such as stationary iterative ones, Krylov subspace ones, pivoting ones, and non-smooth Newton ones [see 2, for a detailed list and references]. This makes comparison difficult. As for testing a solver during development, reference cases are necessary and should be easily accessible from a prototyping environment, where one can perform tests on the incremental problem defined in discussed in Sec. 2 first. These issues have been discussed previously [6, 8, 14]. But since there isn’t an exchange format yet, one is restricted to work on simplistic examples, or random problems [8, 14], which is hardly representative of real scenarios.

Benchmarking of multibody dynamics simulation library is already being addressed [12, 18]. But the focus in the latter studies and benchmark frameworks is to evaluate the global performance of a numerical method on given problems defined by configuration, initial values, and scenario. The
test results are then affected by the overall formulation, the time integration algorithms, and the numerical methods used for solving forces and accelerations as well. As useful as this is, we are concerned with a different aspect of benchmarking.

Solver evaluation is concerned only with the computation of forces and accelerations for the incremental problem, needed at any one stage of a time integration method. It is thus useful to have access to numerical kinematic data instead of having to write a simulation library or linking to one, or resort to random data. With raw data, one can work with scripting languages and their libraries, such as MATLAB, OCTAVE, or PYTHON to name but three. This is the strategy we present here. Our datasets contain numerical values of mass matrices, Jacobians, velocities, forces, etc., for the descriptor form of the equations of motion. Minimal coordinate extensions are forthcoming. This data can then be processed easily by the aforementioned software tools without writing any interfacing software.

The data storage involves only sparse matrices in packed formats, vectors, and scalars, which are all naturally supported. This can be rearranged and adapted to the need of any solver, existing ones in particular, which can then be used via plugins for the tools mentioned already. Rearrangement includes transforming to dense matrices from sparse ones, using row or column major formats, using saddle or Delassus operator based formulations, for instance. We chose HDF5 since this is natively supported by a plethora of data analysis and scripting software suitable for numerical computations.

A large collection of reference problem sets can then be generated easily with only a few simulation libraries which can write in this format, or whose output files can be converted to same if all the data is available. This is more general than declarative exchange formats in the present context, since it is irrelevant at the numerical level whether a given Jacobian corresponds to a screw or tripod joint. Only numerical values are provided, and that all a solver needs to know for the incremental problem. The same applies for contact computations for instance, as not all libraries have support for all types of geometries, or generate contact information in the same way. No special software is needed to access the problem sets, and not everyone has to adapt their software libraries.

By limiting the contact descriptions to points and normals, the user is free to choose the representation of contact manifolds. Any formulation of the contact problem can be expressed this way, any friction law can be used, and the data layout can be adapted to any type of solver. The restriction is that we use the descriptor form. This contrast with a similar effort FCLIB [1], but which concentrates on a specific problem formulation in a specific matrix form. This is ill-suited for many solvers, such as stationary iterative methods for instance, which need original contact information, unavailable in FCLIB.

Performance metrics are needed to develop better solvers and are currently lacking. These metrics must match computed solutions against the chosen physical model in a detailed, comprehensive way, not limited to global errors which is the most commonly used now. Metrics are also needed to provide a global statistical picture of the quality and performance of a solver on a large number of problems, and visualization tools or templates are needed to present this information.

The rest of the paper covers the definition of the incremental problem in Sec. 2, and the specification of the datasets in Sec. 3. The analysis pipeline is introduced in Sec. 4, with simple usage examples. Quality metrics are defined in Sec. 5. We present experimental results using our tools in Sec. 6 with discussion in Sec. 7 and conclusions in Sec. 8.

2 PROBLEM DEFINITION

We consider the incremental problem of computing constraints and contact forces for a planar or spatial multibody system subject to dry frictional contacts. This is solved at any stage of a numerical time integration method. We concentrate on the spatial case in what follows for brevity.
In descriptor form, this can be formulated as follows:

\[
\begin{align*}
Mv &= G^T \lambda + N^T \nu + D^T \beta + a \\
Gv &= b, \\
Nv &\geq c, \quad \nu \geq 0, \quad \text{and} \quad \nu \cdot (Nv - c) = 0, \\
\mathcal{F}(\nu, \beta) &= \text{true},
\end{align*}
\] (1)

Bold face is used for vectors and matrices, the latter in upper case, \((\cdot)^T\) is the transpose operator. Definitions follow.

The unknowns are the velocity vector \(v\), the multipliers \(\lambda\), \(\nu\), and \(\beta\) correspond to constraint, normal and tangent forces, respectively, according to \(G^T \lambda\), \(N^T \nu\) and \(D^T \beta\), respectively. The unknowns are to be computed so they satisfy a chosen friction law, \(\mathcal{F}(\nu, \beta) = \text{true}\), with any numerical method chosen by the user of the dataset.

Quantities contained in or derived from the datasets are as follows.

The system’s mass matrix \(M(q)\) is in represented in inertial frame, including inertia tensors, and depends on the generalized coordinates \(q\) which include quaternions for rotations. The generalized velocity \(v\) is expressed in Cartesian coordinates, and includes translational and angular components.

The Jacobian matrix \(G\) corresponds to locally linear equality constraints \(g(q) = 0\) so that \(\dot{g}(q) = Gv\). The constraint definition is irrelevant for the computation of the incremental problem, but we do provide the value of \(g(q)\) in the data file, allowing for constraint violation.

Vectors \(a, b,\) and \(c\) depend on the chosen time discretization and stepping model, and are computed from the velocities, gaps and other information in the dataset according to the chosen model.

Matrix \(N\) projects the generalized velocities onto the normal bundle of the manifold \(\chi(q) = 0\), and \(D\) projects to the tangent bundle of same.

The complementarity condition on the third line of Eqn. (1), where inequalities are understood component-wise, means that separating contacts corresponds to components \(\chi^{(j)}(q) > 0, j = 1, 2, \ldots, n_{\text{contacts}}\) have zero normal forces. Resting contacts with \(\chi^{(j)}(q) = 0\) have positive normals. Since penetration \(\chi^{(j)}(q)\) can occur due to numerical errors, the vector \(c\) is included for generality, and would be computed and used to include constraint stabilization according to a given stepping scheme.

The definition of the friction law, \(\mathcal{F}(\nu, \beta)\) is left to the user.

We believe that the incremental problem defined in Eqn. (1) covers all models expressed in the descriptor form for linearized constraints.

We left out joint friction and rheonomic constraints in Eqn. (1) for brevity but these are supported by the dataset specification.

Friction models based on position projection such as the Paoli-Schatzman scheme [20] are based on three step recurrences

\[q_{k+1} \leftarrow \Phi(q_k, q_{k-1}),\]

where \(k\) is the discrete time and \(\Phi\) is a mapping. This is also compatible as long as it is possible to invert the mapping \(\Phi\) using the coordinates \(q_k\) and velocities \(v_k\). The datasets contain enough information to represent these as well.

3 DATASET SPECIFICATION AND HANDLING

We previously described the layout of the required kinematic data in the Hierarchical Data Format 5 (HDF5) [17] and started to collect problem sets on our webpage http://tinyurl.com/rpiih5. An almost complete description of the hierarchical specification implemented with HDF5 is de-
scribed in Fig. 1, excluding the optional, non-essential data. A more technical description is published on our website.

The Jacobian matrix $G$ is stored explicitly in block-packed format, with two blocks per constraint, one block per constrained body. The number of rows is the number of individual equations, e.g., three for a ball joint, five for a hinge. An extension in preparation will cover constraints with more than two bodies. There is no reference to the type of joint it is associated with.

Body variables such as mass, inertia tensor, position, velocity, angular velocities and forces are stored body-wise. This means that assembly is necessary if one wants the corresponding system-wide variables. The reason is that stationary iterative methods as well as Krylov subspace ones operate directly on individual bodies and individual constraint Jacobian blocks or individual rows. There lies one of the advantages of this specification when compared to that of FCLIB [1].

The normal and tangent Jacobians, $N$ and $D$, are not stored at all in the data file as illustrated in Fig. 1. Instead, the user constructs these from contact points and normals according to the chosen representation of the contact manifold $\chi(q) = 0$. This allows for point-wise or surface-wise models. Because of this, one can, for instance polygonize the Coulomb friction cone [22], or include the Contensou effects [15] for surface models. The contact data also allows for geometric analysis, for contact reduction for instance. There is a limitation however since the original geometries are not included. That would over-complicate the data specification.

Optional contact information includes tangent vectors $t_u, t_v$ which span the local contact surface which can be used to model anisotropic friction, in which case additional friction coefficients can be stored.

The generality of this specification is discussed further in Sec. 4.

The datasets comes from simulations which write “frames” – kinematic data – at selected time steps. A file can contain datasets from any number simulations, and each simulation containing any number of frames, each frame containing the data for one time step. Though a frame corresponds to the state given by one library, one solver, this is nearly irrelevant when one considers the incremental problem at the numerical level. The accuracy at which the frame was computed for instance does not affect the validity of the data itself. The datasets are then loaded in the analysis environment and assembled in suitable form to be processed by a solver. Also included is a “manifest” for each simulation which describes the origin of said, along with any information relevant to reproduce same, though only a few descriptive words are required. An image can and should also be inserted. We are now defining keywords which should be present for use in automatic generation of plots. This layout makes it easy to curate datasets since few files are involved and since they are self-describing.

Instrumenting an existing simulation library is relatively simple. We provide a minimal C++ interface to HDF5 which hides the complexity of the standard HDF5 API for the needs at hand. We
also provide a plugin for ODE [21]. Since ODE is integrated in Gazebo [11], widely used for robotics simulations, and is also integrated with V-Rep, also designed for robotics, a multitude of datasets can be generated with very little effort. Datasets can also be produced with AgX, a commercial multibody dynamics code [4], as well as the RPI-Matlab Simulator [23]. Other libraries can presumably store this kinematic data in which case conversion to our format requires straightforward scripting. We understand that introducing HDF5 write functionality in an existing library is intrusive, be it easy, but conversion from existing formats should be possible. In the worst case scenario, one would use an instrumented simulation library to obtain data for an example of interest not yet in the collection.

We have already produced many datasets covering a variety of configurations, including systems with joints and simpler stacking ones of different sizes.

The model we emulate here is that of the University of Florida Sparse Matrix Collection [5], a collection with far more consumers than producers.

4 ANALYSIS PIPELINE

We provide scripts to assemble the data according to the most commonly used formulations and friction laws. From that point, any solver which has an interface for or is written in MATLAB directly, is very easy to test. The same goes for OCTAVE. We wrote scripts to produce GAMS files using the GDXMRW utilities, and this alone opens the door to a large number of solvers designed for general problems in mathematical programming. We wrote plugin code to link solvers found in the Siconos platform http://tinyurl.com/siconos, as well as for the Lemke [16] solver found in MOBY [7] among others. We also provide script versions of solvers we wrote ourselves. All this software is posted on our website under permissive licenses. Python offers extensive HDF5 support so it should be easy to do the same.

As a usage example, one can, for instance, construct the LCP corresponding to the polygonized friction cone [22] from the dataset found in a frame shown in Fig. 1, with a simple command:

\[
[A, q] = h5\_build\_stap\_reduced (frame, n\_polygon\_sides);
\]

yielding a matrix \(A\) and a vector \(q\) corresponding to the LCP

\[
0 \leq z \perp Az + q \geq 0.
\]  

What the function does is construct all matrices in Eqn. (1) but with a special definition of \(N\) and \(D\) which is related to the polygonization, compute Schur complements, and assemble \(A\) in a suitable form.

A complete test of a new lemke [16] algorithm on a simulation dataset would read as follows.

```matlab
pivots = []; metrics = cell();
p = load('problems.h5');  \% file problems.h5 contains a dataset
for [ frame, key ] = p.simulation_00001;
    \% sol contains number of pivots and complementarity error
    [sol] = moby\_lemke(A, q);
    pivots = [pivots; sol.pivots];
    metric = coulomb\_metrics(frame, sol, n\_polygon\_sides);
    metrics{end+1} = metric;
end
```

And here for instance, the sparse matrix \(A\) would be processed in the \texttt{mex} file \texttt{moby\_lemke} and formatted suitably for the Lemke solver in the MOBY [7] library, a dense, row major matrix in this case. After this, the data goes back through the \texttt{mex} file, information is packed into a struct. The \texttt{coulomb\_metric} script will then analyze the solution and compute the different errors mentioned below in Sec. 5, as they apply this friction law. One would presumably keep track of the iterations,
the number of pivots in this case. This simple script would then provide data for a large set of problems and potentially reveal anomalies not present in simple or random problems, but which would appear later in situ, at which point they are hard to isolate and understand.

5 QUALITY METRICS AND STATISTICAL ANALYSIS

A good solution to a frictional problem should satisfy the friction law in all its details. The global error averages all errors and provides too little information. For instance, if a contact point is reported as sticking, the tangential velocity should be zero. The error for this should be the residual tangential speed or creep, and this should identified separately. For a sliding contact, one needs to know whether or not the friction force opposes the velocity. To our surprise, it is often the case that the computed friction forces are aligned with the sliding velocity. A measure of error here is the misalignment between the two. One also needs to know whether or not the transition between stick and slip modes is captured properly. Penetration is nearly inevitable and this too deserves its own measurement.

It is also important to understand the difficulty of a given problem. It is our experience that problems involving sliding contacts are more difficult to solve and so we measure both the total number of resting contacts as well as the fraction of sliding ones.

We measure the following:

- total number of resting contacts
- number of sliding contacts
- global error as defined by a given solver
- nonpenetration error: \( \| \chi + hNv \|_2 \) in the two norm, and \( h \) is the time step
- slide alignment error for each contact: \( \frac{1}{\| \tilde{v}(j) \|} \left| \frac{t(j)(\tilde{v}(j))}{\| \tilde{v}(j) \|} + 1 \right| \), and the two norm of these
- stick residual velocity error for static contacts: \( \| \tilde{v}(j) \| \) and the two norm of these
- cone satisfaction error: \( | \min(0, \mu(j)^j \nu(j) - \| f(j) \|) | \), and two norm of these
- anomalous friction contacts: \( \max(\text{sgn}(f(j) \cdot \tilde{v}(j)), 0) \), and the sum of these,

where \( \text{sgn}(x) \) is the sign function.

Other metrics are related to the performance of different types of solvers as well as statistics for global evaluation on very large sets of problems are not presented here but are forthcoming.

6 EXPERIMENTS

We chose only two experiments for illustrative purposes. Comprehensive analysis is beyond the scope of the present article.

We extracted problems from a simulation in which logs are dropped and pile up on an inclined plane at a sufficient angle to cause sliding. We also include results from a simulation of a wheel loader shoveling rocks, and this is typically used in operator training. Still frames are shown in Fig 2. For the logs example, we inclined the plane to guarantee the presence of sliding contacts which, as discussed previously, leads to more difficult problems are results show.

The data was collected from the AgX toolkit www.algoryx.se first. This uses a split solver, where standard constraints and normal forces are computed with an block principal pivot LCP solver [13], and friction are computed with PGS.

We then used our framework to experiment with four different solvers on the slanted log pile. First comes PGS, then a solver of Morales et al. [19] using a combination of subspace minimization
method and PGS, the non-smooth Newton method of Alart and Curnier [3], and the AgX solver itself. These are labeled as “PGS”, “Morales”, “Alart-Curnier”, and “Block Pivot”, respectively, and appear anticlockwise from the south-east corner in Fig. 3, on which the metrics of Sec. 5 are plotted. We let the PGS solver reach a stagnation point which is around 200 iterations. For the other solvers, we chose parameters so they would perform approximately the same amount of work in terms of linear algebra. But for these solvers as well, the solutions produced appear to be the best they can deliver according to our experiments.

We arranged the results in such a way as to give a global picture of the solver’s performance over many frames. The histogram is cumulative: the height corresponds to the total number of contacts. The number of contacts with bad sliding direction appear at the bottom in black, above, in light red is the number of sliding contacts with good alignment, and in dark red at the top, the number of sticking contacts. The alignment error is the red line, creep is in pale blue, and the friction cone violation is in black. There is one column and point on each curve for each frame extracted from the simulation. Clearly, alignment errors are big and the number of cases of “negative” friction is significant for all solvers, all configurations. Given that not all contacts are sliding, this would not be noticed in the global error. Worthy of notice is that the total number of contact varies, meaning that solvers identify different numbers of resting and separating contacts. The Morales solver underestimates the former if the other three solvers are any indication of the correct solution. The Alart-Curnier solver exhibits a number of glitches, even though the problems are similar from frame to frame.
For the wheel loader simulation, we only present the result from the AgX solver in Fig. 4. The PGS solver failed systematically on this problem. The Morales solver is not designed to handle problems with both constraints and contacts. And the Alart-Curnier solver was too slow and had too many failures. It is not clear if this is due to an implementation error, or because of numerical issues such as ill-conditioning. What the graph demonstrates is that errors are consistent, but the amount of creep is significant, so is the number of contacts with negative friction. This is problematic for a vehicle as this causes the wheels to slip. This information can be used then to improve the situation by testing parameters in the analysis environment, i.e., without running the simulation again.

![Graph](image)

Figure 4: Performance statistics for the block pivot solver on the wheel-loader.

7 DISCUSSION

The paper is not an article about solvers and we present data only to show that the framework can give extensive quantitative information about given numerical methods. Yet the data shows clearly that at least four methods which have gained some acceptance and are used routinely demonstrate problems of the kind which has seldom if ever been reported before. Our point and our contribution is that out dataset specification makes it easy to test and compare solvers, and the metrics we introduced provide comprehensive information allowing for deeper understanding.

8 CONCLUSION

We believe that the HDF5 layout specification, the metrics we introduced, along with the datasets and software we published on our website can and will help analyzing and constructing new, better and faster solvers, concentrating on algorithms instead of software development. From our own experience in developing solvers, this type of infrastructure is in fact a necessity. Testing new ideas from within a simulation library and running series of test cases is much more difficult, and not necessarily more enlightening. There is already a lot of useful information in the solution of a single incremental problem.

We are developing a set of reference datasets which can be used to test the fundamental properties of a solver such as stick-slip transitions, isotropy and creep. A few of these are already known [6] but we believe this is incomplete. Various issues only appear when considering objects with different shapes and scales, and others appear only for very large problems. Ill-conditioning for instance can make a solver fail so such examples must be included in a comprehensive collection. This is why we provide datasets from large problems, allowing to stress test solvers.
ACKNOWLEDGEMENTS

Part of this research was conducted using the resources of High Performance Computing Center North (HPC2N), and support from Algoryx Simulation AB.

This work was partially supported by the National Science Foundation through the National Robotics Initiative NSF CCF-1208468, DARPA W15P7T-12-1-0002, and Lockheed-Martin Advanced Technology Laboratory. Any opinions, findings, conclusions, or recommendations expressed herein are those of the authors and do not necessarily reflect the views of the funding agencies.

REFERENCES


Benchmarking of augmented Lagrangian and Hamiltonian formulations for multibody system dynamics

Francisco González*, Daniel Dopico*, Roland Pastorino†, Javier Cuadrado*

*Laboratorio de Ingeniería Mecánica
University of La Coruña
Mendizábal s/n, 15403 Ferrol, Spain
[f.gonzalez, ddopico, rpastorino, javicuad]@udc.es

† PMA division,
Department of Mechanical Engineering
KU Leuven
Celestijnenlaan 300b, 3001 Leuven, Belgium
roland.pastorino@mech.kuleuven.be

Abstract

Augmented Lagrangian methods represent an efficient way to carry out the forward-dynamics simulation of mechanical systems. These algorithms introduce the constraint forces in the dynamic equations of the system through the use of a set of multipliers. While most of these formalisms were obtained using the system Lagrange’s equations as starting point, a number of them have been derived from Hamilton’s canonical equations. Besides being efficient, they are generally considered to be very robust, which makes them especially suitable for the simulation of systems with discontinuities and impacts. In this work, we have focused on the simulation of mechanical assemblies that undergo singular configurations. First, some sources of numerical difficulties in the proximity of singular configurations were identified and discussed. Afterwards, several augmented Lagrangian and Hamiltonian formulations were compared in terms of their robustness during the forward-dynamics simulation of two benchmark problems. The effect of the formulation and numerical integrator choice and parameters on the simulation performance was also assessed.

Keywords: Multibody system dynamics, augmented Lagrangian methods, singular configurations

1 Introduction

Forward-dynamics simulation of multibody systems is a relatively new area in the field of Mechanics. The progress in computer architectures and software tools during the last decades has boosted both research and industry applications of this technique. Real-time applications such as Human- and Hardware-in-the-Loop (HiL) setups are especially demanding in terms of both efficiency and robustness. As a consequence, a considerable effort has been made within the multibody community to develop fast and reliable simulation algorithms to satisfy these requirements.

Generally speaking, multibody systems consist of a set of rigid or flexible links interconnected by joints. The consideration of the kinematic constraints introduced by the latter usually leads to the need for expressing the dynamics equations as a system of Differential Algebraic Equations (DAE’s). Different approaches can be used to solve such a system, among which Lagrange’s multiplier method is a widely used one [1].

If a mechanical system is described with a set of \( n \) generalized coordinates \( \mathbf{q} \), subjected to \( m \) holonomic kinematic constraints \( \Phi \), the equations of motion can be expressed as

\[
\begin{align*}
\mathbf{M} \ddot{\mathbf{q}} + \mathbf{c} &= \mathbf{f} + \mathbf{f}_c \quad (1a) \\
\Phi(\mathbf{q},t) &= 0 \quad (1b)
\end{align*}
\]

where \( \mathbf{M} \) is the \( n \times n \) mass matrix, \( \mathbf{c} \) contains the Coriolis and centrifugal forces, and \( \mathbf{f} \) and \( \mathbf{f}_c \) are the applied and constraint forces, respectively. Following a Lagrangian approach, the generalized constraint reactions can be expressed as \( \mathbf{f}_c = -\Phi_q^\mathbf{q} \lambda \), where \( \Phi_q = \partial \Phi / \partial \mathbf{q} \) is the \( m \times n \) Jacobian matrix of the constraints and \( \lambda \) is a set of \( m \) Lagrange multipliers.

One of the first augmented Lagrangian algorithms for multibody dynamics was introduced by Bayo et al. in [2]. The proposed method combined a penalty representation of the constraint
forces with an iterative update of the Lagrange multipliers. An extension of the method to handle nonholonomic constraints was also included in [2]. Subsequently, several related methods based on the augmented Lagrangian approach have been developed and can be found in the literature. An implementation of the algorithm in [2] aimed at real-time efficiency was published in [3]. In [4] and [5] mass-orthogonal projections were used together with the augmented-Lagrangian formulation to ensure the satisfaction of the kinematic constraints. These two papers included index-3 versions of the algorithms as well, in which the dynamic equations were combined with the numerical integrator formulas. The resulting system of DAE’s was solved in an iterative fashion following a Newton-Raphson scheme, thus improving the robustness of the method. The original algorithms in [4] and [5] were designed for holonomic constraints alone; an index-3 augmented Lagrangian algorithm able to deal with nonholonomic constraints was later described in [6]. The above mentioned formalisms and other similar ones have been successfully used in the study and simulation of a wide variety of mechanical systems. Application examples include heavy machinery simulators [7], biomechanics [8], and co-simulation settings for vehicle dynamics [9].

It is also possible to obtain the dynamics equations using Hamilton’s canonical equations as starting point. Following this approach, the equations of motion become a system of first order Ordinary Differential Equations (ODE’s) of size 2n, instead of a system of n second order ODE’s. Augmented Lagrangian algorithms based on Hamilton’s canonical equations can also be found in the field of multibody dynamics, e.g. [10], [11]. It was stated in [11] that the methods based on canonical equations are more robust than their classical augmented Lagrangian counterparts and ensure a better satisfaction of the kinematic constraints. This was supported by the performance comparison of two formulations, one representative of each approach, in the dynamic simulation of mechanical systems with singular configurations. Although both algorithms were able to deal with the test problems, the Hamiltonian one did not show pathological behaviour in any of the simulations carried out by the authors. However, these formulations have received comparatively less attention in the literature since they were first presented to the multibody community. Naudet et al. [12] developed a recursive algorithm based on canonical momenta, although they did not follow an augmented Lagrangian approach. The authors affirm in [12] that a possible reason why Hamiltonian equations are rather infrequent in multibody applications is that they are computationally intensive to construct and they cannot compete with acceleration based algorithms, especially recursive ones. More recently, Malczyk et al. [13] combined the Divide and Conquer Algorithm (DCA) with Hamilton’s canonical equations to obtain a parallel algorithm. Their preliminary results suggested that their Hamiltonian approach can outperform the Lagrangian one in terms of accuracy in the enforcement of kinematic constraints and conservation of the mechanical energy of the system.

In this paper, a performance study of several existing augmented Lagrangian and Hamiltonian methods for multibody dynamics is presented. Special attention was paid to their behaviour in the proximity of singular configurations. The comparison of the different algorithms was done using test examples from the IFToMM benchmark problem library [14].

2 Augmented Lagrangian formulations
Several formulations were selected for this study among the many available in the literature. The ones described in [2], [11], and [4] were chosen because they share a similar structure of the dynamics equations. In the following, natural coordinates [15] are assumed to be used in the modelling. This has two important consequences. First, term $c$ vanishes from the dynamics equations. Second, the mass matrix $M$ becomes constant, and so all its derivatives are zero.

2.1 Penalty formulation
Even though it is not an augmented Lagrangian one, it is convenient to briefly describe here the penalty formulation introduced in [2]. This formulation replaces the kinematic constraints $\Phi$ with
penalty spring-damper-mass systems. This is achieved introducing fictitious potential and kinetic energy terms in the integral action $A$ of the mechanical system, as well as a set of dissipative forces. The constraint reactions are then replaced by forces proportional to the constraint violations at the acceleration, velocity, and configuration levels

$$\lambda = \alpha (\Phi + 2\xi \omega \Phi + \omega^2 \Phi)$$  \hspace{1cm} (2)$$

where $\alpha$ is the penalty factor, and $\xi$ and $\omega$ are Baumgarte’s stabilization parameters [16]. Together with the velocity- and acceleration-level expression of the kinematic constraints $\Phi$

$$\Phi = \Phi_q \dot{q} + \Phi_r = 0$$  \hspace{1cm} (3)$$

$$\dot{\Phi} = \Phi_q \ddot{q} + \Phi_r \dot{q} + \Phi_r = 0$$  \hspace{1cm} (4)$$

where $\Phi_r = \partial \Phi/\partial t$, Eq. (2) allows for the transformation of the system of DAE’s (1) into a system of $n$ second order ODE’s

$$(M + \Phi_q^T \alpha \Phi_q) \ddot{q} = f - \Phi_q^T \alpha (\Phi_q \dot{q} + \Phi_r + 2\xi \omega \Phi + \omega^2 \Phi)$$  \hspace{1cm} (5)$$

Terms $\alpha$, $\xi$, and $\omega$ are $n \times n$ matrices in the general case but for simplicity they are treated as scalars in this document.

2.2 Index-1 and index-3 augmented Lagrangian formulations

The penalty formulation in Eq. (5) has the disadvantage of being very sensitive to the value of the penalty factor $\alpha$ in terms of convergence. Additionally, a certain violation of constraints is required to develop the necessary reaction forces $f_r$, so a complete fulfilment of the constraints can never be achieved. The augmented Lagrangian formulations proposed in [2] and [4] intended to overcome these limitations. The Lagrange multiplier method was applied to the solution of Eq. (5) to obtain the index-1 iterative algorithm

$$(M + \Phi_q^T \alpha \Phi_q) \ddot{q} + \dot{\lambda}^* = f - \Phi_q^T \alpha (\Phi_q \dot{q} + \Phi_r + 2\xi \omega \Phi + \omega^2 \Phi)$$

$$\lambda^*_{i+1} = \lambda^*_i + \alpha (\Phi + 2\xi \omega \Phi + \omega^2 \Phi)$$  \hspace{1cm} (6)$$

where $\lambda^*$ are the $m$ modified Lagrange multipliers and subscript $i$ stands for the iteration number. If the multipliers are updated only once, then this formulation is equivalent to the penalty one in Eq. (5). Position-, velocity-, and acceleration-level mass-orthogonal projections were also used in [4] to ensure an accurate satisfaction of the kinematic constraints.

In [4] and [5] the augmented Lagrangian algorithm in Eqs. (6) was combined with the Newmark numerical integration formulas [17]

$$\dot{q}_{k+1} = \gamma h \dot{q}_k - \gamma q_k + \left( \frac{\gamma}{2} - 1 \right) q_k + h \left( \frac{\gamma}{2} - 1 \right)$$

$$\ddot{q}_{k+1} = \frac{1}{\beta h^2} q_{k+1} - \ddot{q}_k ; \hspace{1cm} \ddot{q}_k = \frac{1}{\beta h^2} q_k + \left( \frac{1}{2\beta} - 1 \right) \ddot{q}_k$$  \hspace{1cm} (7)$$

where $h$ is the integration step-size, $\beta$ and $\gamma$ are scalar parameters of the integrator formulas, and subscript $k$ denotes the time-step, to obtain an index-3 algorithm with the system generalized coordinates $q$ as primary integration variables. Establishing the dynamic equilibrium at time-step $k + 1$ yields

$$Mq_{k+1} + \beta h^2 \Phi_q^T q_{k+1} \left( \lambda^*_{k+1} + \alpha \Phi_{k+1} \right) - \beta h^2 f_{k+1} - \beta h^2 Mq_k = g(q, \dot{q}) = 0$$  \hspace{1cm} (8)$$

where the assumption that the velocity and acceleration projections remove the constraint violations at those levels has been used. The system of nonlinear equations in Eq. (8) is then solved by means of a Newton-Raphson iterative approach

$$\left[ \frac{d g(q, \dot{q})}{dq} \right]_{q_i} \Delta q_{i+1} = -g(q, \dot{q})_{i}$$  \hspace{1cm} (9)$$

1550
The leading matrix in Eq. (9) can be approximated as [5]

\[
\begin{bmatrix}
\frac{dg(q, \dot{q})}{dq} \\
\frac{dg(q, \dot{q})}{\dot{q}}
\end{bmatrix} \approx M + \gamma \dot{h}C + \beta h^2 (\Phi_q^T \alpha \Phi_q + K)
\]

(10)

where \( C = -\partial f / \partial \dot{q} \) and \( K = -\partial f / \partial q \). The Lagrange multipliers can be updated during the iterative process in Eq. (9) as

\[
\lambda^{i+1}_i = \lambda^*_i + \alpha \Phi_q^T q
\]

(11)

The index-3 augmented Lagrangian formulation (ALi3) described by Eqs. (8)–(11) with velocity and acceleration projections features excellent robustness and efficiency properties and it has been successfully used in real-time simulation of medium-size and large multibody systems [7], [9].

2.3 Formulation based on Hamilton’s canonical equations

Formulations based on Hamilton’s canonical equations constitute an alternative to the classical, acceleration-based augmented Lagrangian algorithms. They introduce the conjugate or canonical momenta \( p = \partial L / \partial \dot{q} \), where \( L \) is the system Lagrangian, as system variables besides the generalized coordinates \( q \) [1]. With the definition of the Hamiltonian \( H = p^T \dot{q} - L \) the canonical equations for a constrained system can be written as [11]

\[
\dot{q} = \frac{\partial H}{\partial p}; \quad -\dot{p} = \frac{\partial H}{\partial q} - f_{nc} + \Phi_q^T \lambda
\]

(12)

where \( f_{nc} \) are the non-conservative forces applied to the system. Following a procedure similar to the one described in [2], an augmented Lagrangian algorithm can be developed from Eqs. (12). Again, a fictitious potential energy term and dissipation forces are introduced in the Lagrangian and a penalty approach is followed to obtain [11]

\[
(M + \Phi_q^T \alpha \Phi_q) \dot{q} = p - \Phi_q^T \alpha \left( \Phi_t + 2\xi \omega \Phi + \omega^2 \int_{t_0}^{t} \Phi dt \right) - \Phi_q^T \Phi
\]

(13)

where \( t_0 \) is the starting time of the motion and \( \Phi \) are the formulation multipliers, which verify \( \Phi = \lambda \). The time derivatives of the canonical momenta can be explicitly obtained from equation

\[
\dot{p} = f + \Phi_q^T \alpha \left( \Phi + 2\xi \omega \Phi + \omega^2 \int_{t_0}^{t} \Phi dt \right) + \Phi_q^T \Phi
\]

(14)

and the multipliers \( \Phi \) are iteratively updated following

\[
\Phi_{i+1} = \Phi_i + \alpha \left( \Phi + 2\xi \omega \Phi + \omega^2 \int_{t_0}^{t} \Phi dt \right)
\]

(15)

The algorithm in Eqs. (13)–(15) can also be used in a penalty fashion if the number of updates of the multipliers in each evaluation of \( \dot{q} \) is set to just one.

3 Rank deficient Jacobian matrices and singular configurations

The application of the Lagrangian approach to the dynamics equations (1), together with the differentiation of the kinematic constraints (1b) with respect to time, results in a system of linear equations that can be written as follows

\[
\begin{bmatrix}
M & \Phi_q^T \\
\Phi_q & 0
\end{bmatrix} \begin{bmatrix}
\dot{q} \\
\lambda
\end{bmatrix} = \begin{bmatrix}
f \\
-\Phi_q q - \Phi_t
\end{bmatrix}
\]

(16)

If the Jacobian matrix \( \Phi_q \) is rank deficient, the leading matrix of system (16) becomes singular. This means that an infinite set of values of the Lagrange multipliers \( \lambda \) are valid solutions of the
system and some additional assumptions must be made to choose one solution among all the possible ones [18]. Rank deficient Jacobian matrices can be the consequence of the presence of redundant kinematic constraints. In this case, the Jacobian matrix is usually rank deficient during the whole motion of the system. Another possibility is the existence of singular configurations in the workspace. When the system reaches one of these singularities, the number of degrees of freedom (DoF) suddenly increases and the Jacobian matrix undergoes a loss of rank.

All the algorithms presented in Section 2 are able to deal with rank deficient Jacobian matrices. The leading matrices in Eqs. (5), (6), (10), and (13) are all symmetric and positive-definite, provided that an appropriate penalty factor $\alpha$ has been selected. The use of the penalty technique is equivalent to assuming a certain stiffness distribution within the system and this reduces the number of valid solutions for $\lambda$ to only one [19]. However, they may still experience numerical difficulties in the proximity of singular configurations.

### 3.1 Benchmark examples

Several multibody systems involving redundant constraints and singular configurations can be found in the IFToMM library of benchmark problems [14]. Among these, we have selected three for the comparison of the dynamic formulations in Section 2. The first one is a six-link rectangular Bricard mechanism (Fig. 1). This is a redundantly constrained, one-DoF mechanical system frequently used as benchmark problem (e.g. [20]). The set of kinematic constraints which are linearly dependent cannot be a priori identified, as it changes during motion. Therefore, redundant equations cannot be simply eliminated from the constraint set $\Phi$, and the Jacobian matrix $\Phi_q$ is permanently rank deficient. However, the system does not reach any singular configuration during its entire range of motion.

![Figure 1: A six-link rectangular Bricard mechanism, a redundantly constrained multibody system without singular configurations.](image)

Two planar linkages were chosen as examples of systems that undergo singular configurations: a slider-crank mechanism and a double four-bar linkage (Fig. 2). These were already used in [11] to discuss the performance of augmented Lagrangian formulations in the simulation of systems with singular configurations. Both are made up of rods of length $l = 1$ m with a uniformly distributed mass $m_b = 1$ kg and a square cross section of width $r = 0.1$ m, connected by revolute joints. Gravity ($g = 9.81$ m/s$^2$) acts in the negative direction of the $y$ axis in the three examples.

The forward-dynamics simulation of the motion of the Bricard mechanism can be used to show that the augmented Lagrangian formulations described in Section 2 are able to successfully deal with rank deficient Jacobian matrices derived from the presence of redundant constraints. Conversely, numerical difficulties were observed during the simulation of the systems in Fig. 2 when they were near a singular configuration.

### 3.2 Behaviour of the formulations in the neighbourhood of a singular configuration

The slider-crank mechanism in Fig. 2a is in a singular configuration when its two rods are aligned on the global $y$ axis. The linkage has one DoF during the rest of its motion, but at this configu-
ration a new degree of freedom instantaneously appears. The singular configuration is in fact a bifurcation point, after which the system can continue its slider-crank motion or start to behave as a simple pendulum with point $P_3$ stopped at the global origin of coordinates. Both motions are actually possible when the linkage is exactly in the singular configuration and momentarily becomes a two-DoF system. The singular configuration for the four-bar linkage (Fig. 2b) happens when all the links are aligned on the global $x$ axis; again, we have a bifurcation point at which two alternative motions are simultaneously feasible.

3.2.1 Change in the subspace of admissible motion in singular configurations

It can be useful to decompose the system velocities into its components contained in the subspaces of admissible and constrained motion [21] to highlight the role of singular configurations as bifurcation points. Given a mechanical system described with a set of $n$ generalized velocities $\dot{q}$, the $m$ kinematic constraints at the velocity level (3) can be used to define the subspace of constrained motion (SCM). The dimension of the SCM is the rank of the Jacobian matrix $\Phi q$, so this subspace will be $m$-dimensional if the kinematic constraints are linearly independent. The subspace of admissible motion (SAM) complements the SCM. The system velocities can be then decomposed into two components as $\dot{q} = \dot{q}_a + \dot{q}_c$ where $\dot{q}_a$ is the set of generalized velocities admissible with the velocity-level constraints in Eq. (3); $\dot{q}_c$ is the velocity set which is not admissible with the constraints, i.e. constraint violations.

The slider-crank example can be modelled with three planar natural coordinates: the $x$ and $y$ coordinates of point $P_2$, $x_2$ and $y_2$, and the $x$ coordinate of point $P_3$, $x_3$. Two kinematic constraints, enforcing constant distances between the tips of the rods, are necessary to ensure the correct motion of the assembly. The corresponding equations at the velocity level are

$$\dot{\Phi}_{sc} = \begin{bmatrix}
2x_2 & 2y_2 & 0 \\
2(x_2 - x_3) & 2y_2 & 2(x_3 - x_2)
\end{bmatrix} \begin{bmatrix}
\dot{x}_2 \\
\dot{y}_2 \\
\dot{x}_3
\end{bmatrix} = \Phi_{sc}^{ic} \dot{q}_c^{sc} = 0$$

(17)

where $\dot{q}_c^{sc}$ and $\Phi_{sc}^{ic}$ are the generalized velocities and the Jacobian matrix of the slider crank with the selected modelling. Let us consider that at $t = 0$ link $P_1-P_2$ is at an angle $\phi = \phi_0 = \pi/4$ with respect to the $x$ axis, and that $\dot{x}_3 = -4$ m/s. At time $t = t_s$ the system reaches a singular configuration, in which $\phi = \pi/2$, $x_2 = x_3 = 0$, and $y_2 = l$ m. For $t < t_s$, the Jacobian matrix $\Phi_{sc}^{ic}$ has rank two and any admissible velocity set can be expressed as

$$\dot{q}_a^{sc} = \eta \begin{bmatrix}
1 \\
-x_2/y_2 \\
x_3/(x_3 - x_2)
\end{bmatrix}$$

(18)

where $\eta$ is a scalar. At $t = t_s$, the system is in a singular configuration, and the Jacobian matrix becomes

$$\Phi_{sc}^{ic} \big|_{t_s} = \begin{bmatrix}
0 & 2l/0 \\
0 & 2l/0
\end{bmatrix}$$

(19)
which is a rank-1 matrix. The SCM for this instant is a one-dimensional subspace. Consequently, the SAM has dimension two. Among the several alternatives to parametrize this subspace a possible one is

\[
\dot{q}_{SC}^{v}\big|_{t_\ast} = \eta_1 \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix} + \eta_2 \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \eta_1 \dot{q}_{a1}^{v} + \eta_2 \dot{q}_{a2}^{v}
\]

where \(\eta_1\) and \(\eta_2\) are scalar parameters. Vector \(\dot{q}_{a1}^{v}\) corresponds to the slider-crank motion of the mechanism, while \(\dot{q}_{a2}^{v}\) represents a single pendulum motion with point \(P_1\) fixed at the origin. The condition \(x_2 = x_3\) makes both branches simultaneously possible, so the velocity vector \(\dot{q}\) of the system can have components along both \(\dot{q}_{a1}^{v}\) and \(\dot{q}_{a2}^{v}\). However, when the system leaves the singular configuration at \(t > t_s\) it reverts to a one-dimensional SAM, which will be either the slider-crank one compatible with \(\dot{q}_{a1}^{v}\) or the simple pendulum motion defined by \(\dot{q}_{a2}^{v}\), depending on how the numerical integration process proceeded at \(t = t_s\).

This reasoning can be generalized to any 1-DoF mechanical system. The introduction of an extra DoF at a singularity momentarily expands the set of admissible velocities, which becomes a linear combination of a velocity vector in continuity with the pre-existing system motion, \(\dot{q}_{a1}\), and a new one \(\dot{q}_{a2}\) which is also compatible with the constraints. Both components are only simultaneously admissible at the singular configuration; at this point \(\dot{q}_a = \eta_1 \dot{q}_{a1} + \eta_2 \dot{q}_{a2}\). After the singularity, one of the components will define the motion and the other one will become a violation of the velocity-level constraints. Augmented Lagrangian formulations based on penalty approaches transform the constraint violations into constraint reactions, as shown in Eq. (2). Accordingly, penalty-based formulations remove the velocity component along the no longer admissible direction by introducing an impact when the system leaves the singular configuration.

![Figure 3: y reaction force at point P1 during motion of the slider-crank mechanism, starting from a singular configuration, (a) for different initial velocities; (b) for different initial configuration-level constraint violations \(\epsilon\).](image)

The forward-dynamics simulation of the slider-crank motion starting from the singular configuration supports the previous statements. As correctly pointed out in the literature (e.g. [11]), the simulation can be started from a singularity because the formulations in Section 2 are able to find a solution for the dynamics equations even with a rank deficient Jacobian matrix. Here, the penalty formulation in Eq. (5) was used with a penalty factor \(\alpha = 10^7\), Baumgarte parameters \(\omega = 10\) and \(\xi = 1\), and the trapezoidal rule as integrator (a particular case of the Newmark formulas (7) with \(\beta = 0.25\) and \(\gamma = 0.5\)), with a step-size \(h = 10^{-3}\) s. First, the initial velocity was made proportional to \(\dot{q}_{a1}^{v}\) by choosing \(\eta_1 = -2\) m/s and \(\eta_2 = 0\). Afterwards, \(\eta_2\) was given different non-zero values and the simulation repeated for each of them. Fig. 3a shows that introducing a component of \(\dot{q}\) along \(\dot{q}_{a2}^{v}\) gives rise to impact forces in the constraint reactions. Numerical experiments with the other formulations described in Section 2 showed the same behaviour. Moreover, the simulation of a 10 s motion of the four-bar linkage (Fig. 2b) confirmed that the obtained reaction force in
the x direction at point $P_0$ featured the same impact forces (Fig. 4a). To obtain these results, the ALi3 formulation with velocity and acceleration projections, Eqs. (7)–(11), was used, with stringent convergence requirements to ensure that the constraint violations at the configuration and velocity levels remained close to machine precision. Similar force spikes can be observed in other publications in the literature (e.g. [22]). It should be stressed that the velocity component along $\dot{q}_{a2}$ cannot be eliminated by the velocity projections at the singular configuration, because it is not a violation of the constraints at that point. As expected, these impacts are not present in the simulation of redundantly constrained mechanisms without singular configurations, as in the case of the Bricard mechanism (Fig. 1).

![Figure 4: (a) x reaction force at point $P_0$ during motion of the four-bar linkage, showing impacts when the system is near a singularity; (b) mechanical energy of the slider-crank mechanism integrated with the forward Euler method ($h = 10^{-5}$ s) and the penalty formulation ($\alpha = 10^9$, $\omega = 10$, $\xi = 1$). A change of branch occurs at $t = 2.17$ s.](image)

The impact forces above described introduce a series of undesirable effects in the simulations if the numerical integrator and the formulation parameters are not properly selected. They generate discontinuities in the mechanical energy of the system, as shown in Fig. 4b. Sometimes they can cause the mechanical system to undergo a change of branch when it leaves the singular configuration. In this case, a discontinuity in the motion takes place and the system velocities after the singularity are no longer in continuity with the pre-singularity motion compatible with $\dot{q}_{a1}$, but with the secondary one defined by $\dot{q}_{a2}$. In extreme cases they may bring about the failure of the simulation.

### 3.2.2 Effect of configuration-level constraint violations

The formulations in Section 2 are rather robust and they are able to handle large impact forces during the pass through singularity. Numerical simulations showed that $\eta_1$ and $\eta_2$ need be of the same order of magnitude for a change of branch to take place in most cases. The exception is the index-3 augmented Lagrangian formulation. For example, starting the simulation of the slider-crank at the singular configuration with $\eta_1 = -1$ m/s and $\eta_2 = -5 \cdot 10^{-5}$ m/s results in a pendulum motion after the singularity, with $\alpha = 10^9$ and a step-size $h = 10^{-3}$ s. Such values of $\eta_2$ are usually not reached in practice because the velocity projections keep this component small during most of the motion.

A configuration-level constraint violation, however, alters the expression of the Jacobian matrix $\Phi_q$ and modifies the definition of the constrained and admissible subspaces. A modification of the generalized coordinates not compatible with the constraints, $\epsilon$, makes the Jacobian matrix become $\tilde{\Phi}_q = \Phi_q (q + \epsilon)$. In general, $\tilde{\Phi}_q \dot{q} \neq 0$, even though the system velocities have theoretically correct values. This means that part of the admissible generalized velocities will be treated as velocity-level constraint violations, giving rise to the impact forces described in the previous section. Fig. 3b shows the impact forces in the simulation of the slider-crank mechanism with
the penalty formulation and the same parameters of Section 3.2.1 starting from the singularity. A configuration error was introduced in the initial position by making \( x_2 = -\varepsilon \) and \( x_3 = \varepsilon \). The simulations showed that the effect of configuration-level constraint violations is much more critical than their velocity-level counterparts. For instance, an initial error in the order of \( \varepsilon = 10^{-2} \) m is enough to trigger a branch change with \( \eta_2 = 0 \).

4 Numerical results

The existence of singular configurations is not the result of a deficient modelling or the wrong choice of simulation strategy, but a property of some mechanisms. Even though a simulation algorithm be able to deal with rank deficient Jacobian matrices, the enlargement of the SAM in the singularity points described in Section 3.2.1 remains. In fact, all the methods mentioned in Section 2 have been found to fail near singular configurations in the simulation of the slider-crank mechanism and the four-bar linkage for certain values of their \( \alpha, \xi, \) and \( \omega \) parameters.

The natural motion of a mechanism would keep the continuity of the velocities during the pass through the singularities. In other words, the ideal simulation of the system motion should not introduce impact forces in the reactions at the singular configurations. Conversely, large values of these impact forces may result in discontinuities in the mechanical energy, which can lead to changes of branch or the failure of the simulation if the algorithm is unable to recover from the impact. Keeping low the violation of the kinematic constraints, especially the configuration-level ones, is a way to reduce the magnitude of the impact forces. This is in accordance with guidelines provided in the literature (e.g. [10], [11]). A simulation algorithm based on a penalty approach must therefore meet two requirements: good constraint stabilization, especially at the configuration level, and robustness to withstand impact forces. A correct adjustment of the penalty factor \( \alpha \) and the stabilization parameters \( \xi \) and \( \omega \) is necessary to satisfy these requirements. In both the penalty and the augmented Lagrangian formulations, the constraint reactions are proportional to \( \Phi, \dot{\Phi}, \) and \( \Phi \) as shown in Eq. (2). Increasing the value of \( \omega \) assigns more weight to the configuration-level constraint violations, which is convenient to overcome singular configurations. To achieve a similar effect in the Hamiltonian formulation in section 2.3, the term \( 2\xi \omega \Phi \) in Eq. (15) must have a larger weight than the other terms in the equation.

Table 1: Best performances obtained with each formulation in a 10 s simulation of the slider-crank mechanism motion, for a maximum energy drift of 0.1 J. The forward Euler integration formula was used in all cases.

<table>
<thead>
<tr>
<th>Formulation</th>
<th>( \omega, \xi ) adjustment</th>
<th>( h ) (s)</th>
<th>( \alpha )</th>
<th>( \omega )</th>
<th>( \xi )</th>
<th>elapsed time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Penalty manual</td>
<td>2 \cdot 10^{-5}</td>
<td>10^8</td>
<td>25</td>
<td>1</td>
<td>1</td>
<td>1.53</td>
</tr>
<tr>
<td>Penalty automatic</td>
<td>1 \cdot 10^{-5}</td>
<td>10^6</td>
<td>1.41 \cdot 10^5</td>
<td>0.707</td>
<td>3.08</td>
<td></td>
</tr>
<tr>
<td>Aug. Lagrangian manual</td>
<td>1 \cdot 10^{-5}</td>
<td>10^7</td>
<td>10</td>
<td>1</td>
<td>3.83</td>
<td></td>
</tr>
<tr>
<td>Aug. Lagrangian automatic</td>
<td>1 \cdot 10^{-5}</td>
<td>10^7</td>
<td>1.41 \cdot 10^5</td>
<td>0.707</td>
<td>3.85</td>
<td></td>
</tr>
<tr>
<td>Aug. Hamiltonian manual</td>
<td>2 \cdot 10^{-3}</td>
<td>10^9</td>
<td>0.1</td>
<td>1000</td>
<td>0.02</td>
<td></td>
</tr>
</tbody>
</table>

Table 1 shows the performance of each formulation in a 10 s simulation of the slider-crank mechanism motion. The numerical experiments were carried out in an Intel Core i7-4790K at 4.00 GHz. The single-step explicit forward Euler formula was used as integrator. For the penalty and the index-1 augmented Lagrangian formulation, the \( \omega \) and \( \xi \) were automatically set to \( \frac{1}{\sqrt{2}} \) and \( \omega = \frac{\sqrt{2}}{h} \) in a first approach [23]. These parameters were subsequently tuned to improve the simulation efficiency. This proved to be a time-consuming process with the penalty formulation, as energy conservation is noticeably affected by changes in the formulation parameters. On the other hand, the augmented Lagrangian method showed a much more consistent behaviour for a
wider range of the parameters. The parameters of the Hamiltonian formulation were manually adjusted to penalize the configuration-level constraint violations at least 200 times more than the other terms in Eq. (15). Results showed that the augmented Hamiltonian formulation described in Section 2.3 can be one or two orders of magnitude faster than the penalty or the index-1 augmented Lagrangian formulation in the studied case.

Table 2: Best performances obtained with each formulation in a 10 s simulation of the slider-crank mechanism motion, for a maximum energy drift of 0.001 J. The selected integrators were the forward Euler method (FE) and the trapezoidal rule (TR).

<table>
<thead>
<tr>
<th>Formulation</th>
<th>Integrator</th>
<th>Tolerance</th>
<th>$h$ (s)</th>
<th>$\alpha$</th>
<th>$\omega$</th>
<th>$\xi$</th>
<th>elapsed time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aug. Hamiltonian</td>
<td>FE</td>
<td>$-2 \cdot 10^{-5}$</td>
<td>$10^9$</td>
<td>0.1</td>
<td>1000</td>
<td></td>
<td>2.10</td>
</tr>
<tr>
<td>Penalty</td>
<td>TR</td>
<td>$10^{-7}$</td>
<td>$10^{-4}$</td>
<td>1000</td>
<td>1</td>
<td>0.33</td>
<td></td>
</tr>
<tr>
<td>Aug. Lagrangian</td>
<td>TR</td>
<td>$10^{-7}$</td>
<td>$10^{-3}$</td>
<td>200</td>
<td></td>
<td>0.06</td>
<td></td>
</tr>
<tr>
<td>Aug. Hamiltonian</td>
<td>TR</td>
<td>$10^{-7}$</td>
<td>$2 \cdot 10^{-3}$</td>
<td>1000</td>
<td>0.1</td>
<td>0.13</td>
<td></td>
</tr>
<tr>
<td>ALi3</td>
<td>TR</td>
<td>$10^{-5}$</td>
<td>$10^{-3}$</td>
<td>$10^9$</td>
<td></td>
<td>0.07</td>
<td></td>
</tr>
</tbody>
</table>

Next, the simulations were repeated for a maximum admissible energy drift of 0.001 J, as required by the problem definition in [14]. With the exception of the augmented Hamiltonian formulation, it was impossible to meet this requirement using the forward Euler integrator with reasonable step-sizes. The trapezoidal rule was used as an alternative. This integrator introduces an iterative process in each time-step. It was observed that this process may diverge in the proximity of a singularity. This requires the detection of divergence and the interruption of the iteration for the simulation to proceed successfully. Results are summarized in Table 2. Similar results were obtained for a 10 s simulation of the motion of the double four-bar linkage and are shown in Table 3.

Table 3: Best performances obtained with each formulation in the 10 s simulation of the double four-bar linkage motion, for a maximum energy drift of 0.1 J. The selected integrators were the forward Euler method (FE) and the trapezoidal rule (TR).

<table>
<thead>
<tr>
<th>Formulation</th>
<th>Integrator</th>
<th>Tolerance</th>
<th>$h$ (s)</th>
<th>$\alpha$</th>
<th>$\omega$</th>
<th>$\xi$</th>
<th>elapsed time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Penalty</td>
<td>FE</td>
<td>$2 \cdot 10^{-5}$</td>
<td>$10^9$</td>
<td>30</td>
<td>1</td>
<td>2.50</td>
<td></td>
</tr>
<tr>
<td>Aug. Lagrangian</td>
<td>FE</td>
<td>$5 \cdot 10^{-6}$</td>
<td>$10^7$</td>
<td>10</td>
<td>1</td>
<td>12.21</td>
<td></td>
</tr>
<tr>
<td>Aug. Hamiltonian</td>
<td>FE</td>
<td>$10^{-3}$</td>
<td>$10^9$</td>
<td>0.1</td>
<td>1000</td>
<td>0.07</td>
<td></td>
</tr>
<tr>
<td>Penalty</td>
<td>TR</td>
<td>$10^{-7}$</td>
<td>$5 \cdot 10^{-3}$</td>
<td>$10^9$</td>
<td>25</td>
<td>0.04</td>
<td></td>
</tr>
<tr>
<td>Aug. Lagrangian</td>
<td>TR</td>
<td>$10^{-7}$</td>
<td>$5 \cdot 10^{-3}$</td>
<td>$10^9$</td>
<td>20</td>
<td>0.05</td>
<td></td>
</tr>
<tr>
<td>Aug. Hamiltonian</td>
<td>TR</td>
<td>$10^{-7}$</td>
<td>$5 \cdot 10^{-3}$</td>
<td>$10^9$</td>
<td>0.1</td>
<td>0.10</td>
<td></td>
</tr>
<tr>
<td>ALi3</td>
<td>TR</td>
<td>$10^{-7}$</td>
<td>$10^{-2}$</td>
<td>$10^9$</td>
<td></td>
<td>0.02</td>
<td></td>
</tr>
</tbody>
</table>

The numerical experiments showed that a robust and efficient performance in the simulation of systems with singular configurations depends not only on the selected dynamic formulation, but also on the numerical integration formulas. For instance, the augmented Hamiltonian formulation shows very good energy conservation properties with an explicit single-step integrator like the forward Euler scheme. This allows one to carry out the simulations with a step-size larger than the one used with the penalty or augmented Lagrangian methods, as it can be seen in Tables 1 and 3. This comparative advantage is lost if the trapezoidal rule is used instead. The ALi3 algorithm with projections of velocities and accelerations showed a very robust behaviour. Possible reasons for
this are the implementation of the numerical integrator in Newton-Raphson form with the general-
ized coordinates as primary variables instead of fixed-point iteration, and the use of projections
to remove constraint violations.

5 Conclusions
Penalty-based Lagrangian methods for multibody system dynamics can deal with rank-deficient
Jacobian matrices but still suffer from numerical difficulties near singular configurations. These
problems give rise to impact forces that can introduce sudden variations of the mechanical energy
and cause the simulation to fail. In this research, benchmark problems were used to compare sev-
eral augmented Lagrangian formulations in terms of their ability to carry out an efficient simulation
while keeping the mechanical energy constant. It was found that the selection of the numerical in-
tegrator plays a key role in this. In particular, iterative integrators may diverge at the singularity,
and so provisions must be made to stop the iteration process if this happens. In all cases, keeping
the constraint violations at the configuration level under a certain threshold was required to obtain
a successful simulation. This constitutes a guiding principle in the adjustment of the formulation
parameters. Additionally, the formulations must be robust enough to deal with large impact forces.
This suggests that implementing the algorithms in Newton-Raphson form can be advantageous in
problems with singularities, which is currently an open line of research.

Acknowledgments
The first author would like to acknowledge the support of the Spanish Ministry of Economy
through its post-doctoral research program Juan de la Cierva, contract No. JCI-2012-12376.

REFERENCES
[2] E. Bayo, J. García de Jalón, and M. A. Serna. A modified Lagrangian formulation for the
dynamic analysis of constrained mechanical systems. Computer Methods in Applied Me-
for real time multibody dynamic simulation in fully Cartesian coordinates. Computer Meth-
ics: Space–state and descriptor methods in sequential and parallel computing environments.
nonholonomic constraint reactions in an index-3 augmented Lagrangian formulation with
velocity and acceleration projections. Journal of Computational and Nonlinear Dynamics,
[8] B. Ruzzeh and J. Kövecses. A penalty formulation for dynamics analysis of redundant me-
chanical systems. Journal of Computational and Nonlinear Dynamics, 6(2):paper 021008,
2011.


Development of a flexible multibody simulation package for in-house benchmarking

Martijn Vermaut, Tommaso Tamarozzi, Frank Naets, Wim Desmet

PMA Division, Department of Mechanical Engineering
KU Leuven
Celestijnenlaan 300, B-3001 Heverlee, Belgium
martijn.vermaut@kuleuven.be, tommaso.tamarozzi@kuleuven.be, frank.naets@kuleuven.be, wim.desmet@kuleuven.be

Abstract

This paper introduces a new object-oriented Matlab research code for flexible multibody simulations. The flexibility of mechanical systems is becoming an important factor in system design and optimization. New flexible multibody formulations are actively being developed by researchers who need a platform in which they can test and benchmark their newly developed concepts and algorithms. This platform has been developed in object-oriented Matlab code and focuses on giving researchers a framework in which they can easily implement novel ideas. It allows researchers to focus on the implementation of their idea and not the entire multibody code around it. This is achieved by keeping the code modular and by adding layers of abstraction between different components in the code, of which an example is derived in this paper. The framework is finally also used to test and benchmark a few different multibody formulations. This shows how easy it is to change formulations that have a large impact on the equations governing the multibody system, but not on the code that constructs these equations.

Keywords: flexible multibody, research code, object-oriented, model order reduction

1 Introduction

Current developments in the field of mechanical engineering are pushing systems to become lighter in order to increase their performance and energy efficiency. By making mechanical systems lighter, they are also becoming more flexible. The dynamic behaviour of mechanical systems due to their inherent flexibility is becoming an important factor in system design, control, and analysis, since it has a non-trivial impact on the system performance, energy efficiency, and reliability.

As a direct result, flexible multibody systems (FMBS) are becoming more prevalent in mechanical system analyses. Recent developments in this field have made it possible to efficiently apply such flexible multibody simulations in a wider range of applications. Flexible multibody formulations are used within commercial software packages that specialize on multibody simulations (e.g. Adams, Simulink, LMS Virtual.Lab Motion, LMS Imagine.Lab AMESim) by adding the effects of a large rigid body motion to a deforming body. The rigid body motion of a body can not be considered as independent from the body’s deformation, which makes it more difficult to add these effects accurately and in a computationally efficient manner.

For these reasons a lot of research effort has been focused on this matter. Some research topics focus on improving the body formulation (e.g. Absolute Nodal Coordinate Formulation (ANCF) [1], Generalized Component Mode Synthesis (GCMS) [2, 3]). Other research topics try to develop Model Order Reduction (MOR) schemes to reduce the computational effort these simulations imply at the cost of a minimal loss of accuracy (e.g. Global Modal Parametrization (GMP) [4], Static Mode Switching (SMS) [5], Parametric Model Order Reduction (PMOR) [6]). Research topics also focus on improving existing solver algorithms in order to adapt them to the challenges that the multibody simulations offer [7].

Research progress in these topics is performed methodologically at first, but as the research progresses, the newly developed abstract concepts need to be implemented, tested, and validated.
Often this phase results in new insights and thus further development of the methodology. This means that the implementation aspect is an essential part of a researcher’s work. Therefore, researchers need a framework to facilitate this as much as possible. The code development itself has been facilitated in recent years by developments in the programming languages themselves (e.g. Matlab, Python-Numpy, etc.) and the programming paradigms (e.g. object-oriented programming support in the aforementioned programming languages). Especially the introduction of object-oriented features in these prototyping languages has allowed a much greater degree of modularity than the functional paradigm previously employed. This implies that different aspects of the multibody simulation can be substituted much more easily by other modules or even used in parallel, without having to touch any other aspect of the code. This can drastically speed up the development cycle of novel multibody formulations.

In order to reduce the time and effort researchers have to spend in developing this full multibody code and in order to be able to test and validate new algorithms using multiple test cases, a multibody research code has been developed. This research code focuses on being as re-usable, as modular, as plug-and-play, and as easy-to-use as possible. It has been developed in Matlab, due to the ease of use and because it offers a lot of tools for researchers to quickly implement new algorithms. This tool has also been developed in object-oriented code, because it allows it to be very modular, and it makes it a lot easier and intuitive for new users. Bodies are actual objects within the code and have separate methods to compute e.g. its mass, or stiffness matrix.

2 General framework structure

As mentioned before, the research code is developed in object-oriented Matlab code. There are four main classes representing the most important entities within the multibody code: bodies, frames, joints, and forces. These classes represent physical objects within the model that is being considered. This makes it easier for researchers to reason about their meaning within the code and to develop code starting from a formula or algorithm.

Bodies represent objects that have Degrees Of Freedom (DOFs) within the multibody systems. They are represented by inertial forces (mass terms) and internal forces (damping and stiffness terms). Both the inertial and internal forces do not need to be linear in the position, velocity, or acceleration of the body DOFs. Take for example the Floating Frame of Reference Component Mode Synthesis (FFRF-CMS) [8] body formulation, which contains non-linear inertial forces and linear internal forces.

Frames represent connection or interface points to the bodies. They create a uniform way of interacting with a body, independent of the body formulation (e.g. FFRF-CMS, ANCF, or GCMS) that is used. When seen from outside a body, frames have a position, orientation, velocity, etc. The body formulation then provides the values to be used for the evaluation of the frame quantities. Objects outside of the body never have to take the body formulation explicitly into account. For the definition of constraints and generalized forces, point Jacobians (i.e. the partial derivative of the point position and orientation to the body generalized coordinates) are needed. The derivation of these Jacobians is given in section 2.2.

Joints represent a set of constraint equations. In a multibody system, multiple bodies are connected to each other through constraints. These constraints introduce algebraic equations to the system which turns the set of equations of motion of the bodies (which are ordinary differential equations (ODEs)) into a set of differential-algebraic equations (DAEs) for the assembled system, which is one of the reasons why system-level MOR is gaining attention. These constraints also use the aforementioned Jacobian of a frame to construct the constraint Jacobian.

Forces are the generalized loads that act upon the body DOFs. Force elements give the force vector in local Cartesian coordinates ($F$), as well as the transformation matrix ($B$) that transforms Cartesian forces to the generalized forces ($Q = BF$) that act on the body DOFs. The transformation matrix is constructed by using the point Jacobian. The tangent stiffness ($K_t = \frac{\partial Q}{\partial q}$), tangent damp-
ing \((C_t = \frac{\partial Q}{\partial q})\), and tangent mass \((M_t = \frac{\partial Q}{\partial \dot{q}})\) matrices of the generalized load can be computed in order to increase the solver accuracy, speed of convergence, and possibly stability, if the solver makes use of these matrices.

2.1 Data flow

The generality of the code is achieved by defining interface points to the bodies, called frames. These interface points are independent from the body formulation when seen from the outside (a point simply has a position, velocity, orientation, etc.), but are internally dependent on the body formulation that is used.

An example of the data flow within the framework can be seen in figure 1. It also depicts the relations between different objects. This example represents a compliant mechanism [9] to which a spring force is attached between two frames (in this case the frames are defined on the same body). The spring force object retrieves the position of each frames (which is dependent on the flexible deformation) and computes the resulting spring force. The force object does not need to know what body formulation is used to compute the position of each frames, it simply defines a force between both.

![Figure 1: Data flow example](image)

2.2 Point Jacobians

Using d’Alambert’s principle, the virtual work that is generated by a force on a body can be expressed w.r.t. the local Cartesian coordinates of the point \(p\) that the force is acting on. From this expression, a coordinate transformation can be written between any two coordinate sets, e.g. the local Cartesian coordinates of a point \(p\) in which we want to express the force on one hand and the body coordinates in which we want to express the motion of the body on the other hand:

\[
\delta x_p^T F = \delta q^T g
\]  

(1)

The forces expressed in the local Cartesian coordinates of a point \(p\) is denoted by \(F\), the virtual displacement in the local Cartesian coordinates of that point \(p\) is denoted by:

\[
\delta x_p = [\delta u_{p,x}, \delta u_{p,y}, \delta u_{p,z}, \delta \theta_{p,x}, \delta \theta_{p,y}, \delta \theta_{p,z}]^T
\]  

(2)
The force expressed in the body coordinates is denoted by $\mathbf{g}$ and is called the generalized force. The virtual displacement in the body coordinates is denoted by $\delta \mathbf{q}$.

Equation 1 can be transformed to:

$$
\delta \mathbf{q}^T \left( \frac{\partial \mathbf{x}_p}{\partial \mathbf{q}} \right)^T \mathbf{F} = \delta \mathbf{q}^T \mathbf{g}
$$

And thus an expression for $\mathbf{g}$ is obtained:

$$
\mathbf{g} = (\mathbf{J}_p)^T \mathbf{F}
$$

in which $\mathbf{J}_p = \frac{\partial \mathbf{x}_p}{\partial \mathbf{q}}$ is called the Jacobian of point $p$.

This expression allows any force expressed in Cartesian coordinates on any point to be transformed to a generalized force on the body coordinates, which makes this transformation very suited to be used in a general, modular framework. This derivation is also independent of the specific body formulation that is used, as long as this point Jacobian can be defined for the different generalized coordinates used for the different body formulations. This means that the generalized forces can be defined using this Jacobian as a black box and the body formulation then has to supply the point Jacobian. This makes it very easy to change the body formulation without changing the force definition, which is very useful in a framework where the goal is to plug in different body formulations.

A similar approach can be followed for the constraint equations and their partial derivatives (the constraint Jacobian). The derivation then yields the same point Jacobian matrix.

3 Comparison with existing multibody software

Many multibody simulation packages are available. Some of them are developed with a commercial goal in mind, others are developed with a more research-oriented goal and some are developed with an open use in mind. Most of these packages have their limitations which make them not as suited for a platform to develop research code on. One of the main limitations of most multibody simulation packages is that they are limited to simulating rigid bodies only. Rigid body dynamics is still a research topic but its level of maturity is very high and most challenges are well understood. A lot of research effort is focussed on the field of flexible multibody dynamics. A simulation package that is only capable of simulating rigid body dynamics is quite limiting in its use for research purposes.

Commercial packages are typically designed to give the user results as fast as possible that are as accurate as necessary. This means that these simulation packages can use some simplifications, because the developers know what influence their assumptions have on the accuracy of the end result and they can thus judge whether or not they are valid or not. When implementing novel algorithms, it may be possible that these simplifications are no longer valid or easy to assess. A framework for research purposes has to start from a basis that is as accurate as possible and not as accurate as necessary. The algorithms that researchers then add, are only limited by the simplifications of the algorithm itself and not by the simplifications of the framework.

Another inconvenience when working with a commercial code, is that they are often developed with a specific body formulation in mind. The code contains functions and routines that are hard-coded for this specific body formulation. When implementing a new formulation in such a code, a lot of the existing code has to be rewritten, since it is not easily adapted for a new body formulation.

Commercial packages are also very often (if not always) closed-source, which means that the internal simulation data is not readily available. This is done in order to protect the algorithms of the commercial package. For research code purposes, researchers might need access to this information (e.g. system mass, damping, and stiffness matrices), which means that the use of such packages is only limited.
There exist some other packages that are aimed at research, but these are often quite complex because they are supposed to span a lot of different research topics. This makes them very widely applicable, but at the cost of added complexity. Some research packages are not designed to be easily extended, but have other goals in mind (e.g., some codes try to simulate models as fast as possible). This additional complexity makes it often difficult for researchers to properly implement their own code. Researchers typically are not too concerned about the implementation of the entire package and how it is structured. The main goal is to efficiently implement their novel algorithms in a multibody code so that they can test, validate, and benchmark it.

Part of the complexity of these packages comes from the programming language in which they are available. Research code is often developed in a programming language that allows researchers to quickly implement an algorithm (e.g., Matlab). Algorithms developed in other languages (e.g., Fortran, C++) are often a lot more computationally efficient, but the development time increases as a trade-off. It is more beneficial to be able to develop novel ideas in a language that minimizes development time, rather than execution time.

4 Applications

The framework is designed to easily implement different body formulations, MOR techniques, and solvers. As an example, a 3d slider crank mechanism is simulated using two different body formulations (FFRF-CMS and GCMS) with a varying amount of flexibility and a system MOR technique (GMP). Table 1 gives an overview of the different system models that are simulated. It lists the amount of system degrees of freedom there are in each model, how many Lagrangian multipliers are used to model the system, and what type of equations are used. All of these systems are simulated using the Generalized-α [11] solver.
Table 1: Models Overview

<table>
<thead>
<tr>
<th></th>
<th>Rigid</th>
<th></th>
<th>Flexible</th>
<th></th>
<th>Fully Flexible</th>
<th></th>
<th>GMP</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FFRF</td>
<td>GCMS</td>
<td>FFRF</td>
<td>GCMS</td>
<td>FFRF</td>
<td>GCMS</td>
<td></td>
</tr>
<tr>
<td># System DOFs</td>
<td>17</td>
<td>22</td>
<td>21</td>
<td>58</td>
<td>30</td>
<td>139</td>
<td>21</td>
</tr>
<tr>
<td># Lagr. Mult.</td>
<td>15</td>
<td>14</td>
<td>15</td>
<td>14</td>
<td>15</td>
<td>14</td>
<td>16</td>
</tr>
<tr>
<td>Equations Type</td>
<td>DAE</td>
<td>DAE</td>
<td>DAE</td>
<td>DAE</td>
<td>DAE</td>
<td>DAE</td>
<td>ODE</td>
</tr>
</tbody>
</table>

Figure 2 shows the schematic representation of the system. The properties of the crank and rod are summarised in table 2. The axis of the slider is positioned 30\textit{mm} to the side and 10\textit{mm} above the axis of rotation of the crank. The slider has a total mass of 1\textit{kg} and is attached to the end of the rod. This is quite a large mass compared to the rest of the system, but it will ensure that the flexibility of the crank is excited in such a way that is has a non-trivial effect on the simulation results. Between 0s and 0.1s, a C1 continuous force is applied to the free end of the crank in order to get the system moving. After that, the free response of the system is simulated up to 1s. The amplitude of this external force can be seen in figure 3. As a measure of comparison for the different simulations, the position and acceleration of the 1\textit{kg} slider is used.

Table 2: Crank and rod properties

<table>
<thead>
<tr>
<th></th>
<th>Crank</th>
<th>Rod</th>
</tr>
</thead>
<tbody>
<tr>
<td>Young’s Modulus ((E)) ([\text{GPa}])</td>
<td>210</td>
<td>210</td>
</tr>
<tr>
<td>Poisson Ratio ((\nu)) ([-])</td>
<td>0.3</td>
<td>0.3</td>
</tr>
<tr>
<td>Density ((\rho)) ([\frac{\text{kg}}{\text{m}^3}])</td>
<td>7800</td>
<td>7800</td>
</tr>
<tr>
<td>Length ([\text{mm}])</td>
<td>80</td>
<td>300</td>
</tr>
<tr>
<td>Cross Section ([\text{mm} \times \text{mm}])</td>
<td>2 \times 4</td>
<td>2 \times 2</td>
</tr>
<tr>
<td>Total Mass ([\text{kg}])</td>
<td>0.004992</td>
<td>0.00936</td>
</tr>
</tbody>
</table>

4.1 Varying body model

The framework allows to easily change the body definition that is used. As an example, the simulation results of the slider motion in the aforementioned 3d slider-crank mechanism is compared for a body formulation based on a FFRF-CMS approach and for a body formulation based on a GCMS approach. Only the body formulation of the crank is varied here, the rod is kept in a FFRF-CMS body formulation. This means that in the GCMS simulations, there is a body based on GCMS, one based on FFRF-CMS and a point mass formulation (3 translational DOFs). The fact that not all bodies have to be defined using the same body formulation is only possible due to the flexibility of the framework and represents one of its main advantages.

4.1.1 Rigid system

First, the results are compared for a rigid system without any modelled flexibility. The results can be seen in figure 4. There is an almost perfect match between both formulations, but there is still a small difference between the results of both simulations. The GCMS formulation has inherent flexibility and the resulting motion is integrated differently, which may contribute to the difference in resulting motion. Researchers can then investigate where this small error comes from exactly. The cause of this difference becomes more apparent as we look at the energy that the external force has put into the system. When the external force no longer acts on the system, the GCMS system contains 99.5\% of the energy of the FFRF-CMS system. This small difference in energy
may result in slightly different resulting motion, as the lower kinetic energy means lower velocities in the system and thus the response of the GCMS system will lag slightly behind the FFRF-CMS system. Researchers can then investigate whether this difference in energy comes from the inherent nature of both body formulations, or whether the solver algorithm plays an important role in this. Without having a framework that allows to simulate the same system with the same properties, the same boundary conditions, and the same excitation, but with a different body formulation for a single specific body, and solve it with the same solver algorithm, researchers would not be able to accurately compare and benchmark different formulations as has been done here.

![Figure 4: Slider Position and Acceleration (Rigid) - FFRF-CMS vs GCMS](image)

### 4.1.2 Flexible system

The same system is simulated again, but with flexibility added to the crank. To model the flexibility, a modeset with only the first 4 free-free dynamic eigenmodes of the crank are used. The lowest free-free eigenfrequency of the crank is 1666 Hz, but the first system dynamic eigenmode will be much lower in frequency. The modeset that is used can be easily changed within the framework as will be done in the next simulation. The results can be seen in figure 5. The difference in acceleration between both body formulations is quite significant. Figure 6 shows a zoomed view between 0.2 s and 0.4 s. In this figure, it becomes apparent that the dominant oscillation (i.e. the first system dynamic eigenmode) is at a different frequency for both simulations. This is because the boundary conditions are not accounted for in the modeset. By using a limited modeset as has been done here, the physical system is badly approximated. Both body formulations use the modeset differently and thus they approximate the physical system differently. As the modeset becomes more and more complete, both body formulations should converge to the same physical solution of the system, as will be shown in the next simulation.
4.1.3 Fully flexible system

The simulation for a system with flexibility is done again, but for this simulation 9 Residualized Inertia Relief Attachment (RIRA) modes [12] are taken into account for the crank in order to correctly model the boundary conditions. The resulting modeset is mass orthonormal. The results can be seen in figure 7. The GCMS system contains less energy (99.4% of the energy of the FFRF-CMS system) after the external forces stops acting on the system, similar to the rigid case. For this simulation, there is quite a good match between both formulations. Between 0.2s and 0.4s the match is very good (whilst it was very poor in the previous simulation). But the match is still not perfect. To get a better match, more dynamic eigenmodes should be taken into account to ensure that both formulations model the same physical reality as much as possible, and the integration time step should be decreased to eliminate a difference in the integration error.
It is well known that the dynamic eigenmodes alone are not capable of modelling boundary conditions very well, which is why extra modes (RIRA modes in this case) are added to the modeset in order to enhance the completeness of the modelled flexibility. By increasing the accuracy of the simulation, both formulations converge to the same solution. The previous simulation results with only dynamic eigenmodes resulted in quite different results for both body formulations. This could lead researchers to believe that there is a mistake in their algorithm. But in fact, in this case the difference was due to how both formulations approximate the same unique solution of the physical system. By increasing the accuracy of the approximation, both body formulations correspond a lot better. This is what is expected in the end, since both formulations model the same physical system. This emphasises again the need for a general framework that supports all of these computations (e.g. RIRA mode computations) to benchmark new algorithms. If researchers implement their own multibody code and only use dynamic eigenmodes to test their algorithms, they might make faulty conclusions about their algorithm.

4.2 Varying system model

The framework also allows to easily implement different system models. As an example, a full multibody model (based on FFRF-CMS) is compared with a GMP reduced model (based on the same FFRF-CMS model) and with a linear state-space model based on the GMP reduced model (i.e. a GMP model without taking the non-linear terms due to the variation of the projection base into account). The state-space model gives a good example of the results one would get by trying to fit a (time-varying) linear model through a highly non-linear multibody system. The system’s flexibility is modelled with the first 4 dynamic eigenmodes of the crank. The results for
the reference simulation (the original system without applying the GMP reduction) and the full GMP simulation can be seen in figure 8. A zoomed view of the acceleration between 0.2s and 0.4s can be seen in figure 9, which also includes the solution for the state-space model. Researchers can then investigate what these results mean. When developing system-level model order reduction schemes such as GMP, researchers need to able to simulate a system in its unreduced form, as well as its reduced form. Both simulations should simulate the same system with the same boundary conditions, so that the only difference is the reduction scheme. To do this, a lot of researchers write their own research multibody code, whilst this framework offers a lot more flexibility than most ad-hoc solutions.
5 Conclusions
A full flexible multibody code for research purposes has been developed in object-oriented Matlab code. The goal of this research code is to give researchers a framework in which they can develop new algorithms in the field of flexible multibody dynamics. The code is designed to be very modular, so that it gives the flexibility to researchers to focus on specific items within the multibody code, without having to worry about what implications their changes have on the rest of the code. This is achieved by using frames as a sort of general interface to bodies.

An example case is simulated using different body and system formulations. Using this example, the advantages of having a general framework have been shown. The framework allows researchers to easily compare the results they obtain using their algorithm with the results they obtain using a tried and tested implementation that is already present in the framework.

6 Acknowledgements
This work benefits from the Belgian Programme on Interuniversity Attraction Poles, initiated by the Belgian Federal Science Policy Office (DYSCO). The IWT Flanders, the Fund for Scientific Research - Flanders (F.W.O.), and the Research Fund KU Leuven are also gratefully acknowledged for their support. The authors also acknowledge the support of ITEA2 through the MODRIO project.

REFERENCES


Evaluation of the Biomechanical Simulator OpenSim on a Multi-Body System Benchmark

L. Tagliapietra, M. Vivian, R. Caracciolo, M. Reggiani

Department of Management and Engineering
University of Padua
Stradella San Nicola 3, 36100, Vicenza, Italy
email: tagliapietra@gest.unipd.it

ABSTRACT
A main challenge in robotic rehabilitation research is to develop devices able to understand user intention and estimate residual motor capabilities with the goal of modulating the provided support. Solving this challenge requires simulation tools able to predict, together with the behavior of the user and the device, also their interaction. However, such tools are still not available or only partially evaluated in prediction of the dynamic interaction emerging from human-machine cooperation. We believe that a multi-level model simulation could be effectively implemented using OpenSim, a well known biomechanical dynamic simulator. Nevertheless, while its efficacy in musculoskeletal simulation is demonstrated by several research work, its use as a simulator of multi-body mechanical systems is still limited. This manuscript presents the implementation with OpenSim APIs of simulating challenging problems that compose a benchmarks suite proposed to evaluate multi-body systems simulation software. We released the developed software as a freely available source code to support other researchers in solving mechanical problem through OpenSim.

Keywords: Multi-body Systems, Benchmark, OpenSim, Dynamic Simulation.

1 INTRODUCTION
In the latest years, neurorehabilitation researchers have proposed novel robotic assistive devices to support physical rehabilitation. Robotic manipulators, automated treadmills [12, 20], and passive or active orthoses [2, 8] are just a few examples of the current state of research. A main challenge for the next generation rehabilitation devices is to improve the interaction with the user. Metabolic energy consumption, muscle myoelectrical activity, compliance in subject’s muscles and joints are important variables that can give insights on the impact of the device on user conditions [4,6,16]. These variables can be measured or estimated when the subject is wearing the real device, giving helpful information to adapt the system to the characteristics and needs of the patients and pursue a more effective personalized rehabilitation treatment. But the design of an effective rehabilitation requires the availability of these variables before the actual construction of the real device. Indeed, knowledge about the human-device interaction allows to compare different design solutions and evaluate which is the most likely to maximize the impact of the treatment. The prediction of the interaction between a subject and an assistive device requires a simulator able to accurately predict the behavior of both the human and the robotic system, together with their interaction. The availability of such a tool would have a huge impact on the design of new system.

We are currently investigating the possibility to use the OpenSim software [5, 14] as a suitable tool. The final objective of our research is the implementation of a multi-level model simulation methodology (Fig. 1), to predict both the subject-specific neuromusculoskeletal function and the mechanics of the robotic assistive device, as well as the emerging symbiotic interaction due to the human-machine cooperation [21]. OpenSim is an open-source simulator developed by the Neuromuscular Biomechanics Lab of the Stanford University. This tool provides a large set of libraries to implement dynamic simulation of human movements. While its reliability to predict human behavior is demonstrated by a large number of studies from the biomechanical community, its use as a simulator of multi-body systems is still limited.
This paper has the objective of assessing the accuracy and precision of OpenSim in simulating a multi-body mechanical benchmark suite. We implemented in OpenSim the five mechanical problems proposed by the Multi-Body System (MBS) Benchmark [10], each one targeting a different simulation challenge. Simulation results demonstrate that OpenSim libraries provide the tools required to implement these challenging problems, achieving high accuracy.

The remaining of the paper is organized as follows: Section 2 presents a short introduction to OpenSim, including the procedure required to develop a dynamic simulation; Section 3 provides a detailed description of the benchmark suite. Section 4 reports the obtained results compared with the reference solutions proposed by the benchmark authors; finally, Section 5 concludes the paper with possible directions of future works.

2 AN INTRODUCTION TO OPENSIM

OpenSim [5,14] is an open source software developed by the Neuromuscular Biomechanics Lab at the Stanford University. Its purpose is to provide free and widely accessible tools for biomechanical modeling, simulation, and analysis. OpenSim provides researchers with an extensive API library that includes most of the tools required to perform dynamic simulations of multi-body systems. Therefore, even if its use within the robotics community is still limited, it could be successfully used to simulate assistive devices while cooperating with their users.

In the following of this section, we will briefly go through the procedure of creating a dynamic simulation to highlight the tools already available in OpenSim APIs. The first step is the creation of a Kinematic Model (KM) defining the chain of rigid bodies connected through joints. OpenSim APIs provide specific calls to define the dynamic properties of each body (mass, center of mass, and inertial matrix) and a new body can be easily created and connected to others, specifying name and properties of the connecting joint or constraint. The joint definition requires to specify position, orientation, and degrees of freedom of the body. Both position and orientation are defined in respect of another body. A ground body, father of the KM must be defined in each simulation. Each body is also associated to a joint and can have a graphical representation.

After creating the Kinematic Model, it is possible to define the Dynamic Model (DM). While the dynamic properties of each body are defined when the body is created and connected to the KM, other properties of the system requires the execution of specific tasks, such as the creation of the contact model and contact forces, the coordinates setup, and the addition of actuators and related controllers. All the classes and methods required to implement these tasks are already implemented in OpenSim APIs. Model movements can be either reconstructed following the evolution of the system coordinates represented by a set of spline functions that interpolate a known movement or driven by actuators. OpenSim provides some basic actuators (e.g. torque and
Table 1. MBS benchmark problems: simulation challenges.

<table>
<thead>
<tr>
<th>Problem Name</th>
<th>Simulation Challenge</th>
</tr>
</thead>
<tbody>
<tr>
<td>A01 Simple pendulum</td>
<td>Example problem (2D)</td>
</tr>
<tr>
<td>A02 N four-bar mechanism</td>
<td>Singular configuration (2D)</td>
</tr>
<tr>
<td>A03 Andrew’s mechanism</td>
<td>Very small time scale (2D)</td>
</tr>
<tr>
<td>A04 Bricard’s mechanism</td>
<td>Redundant constraints (3D)</td>
</tr>
<tr>
<td>A05 Flyball governor</td>
<td>Stiff system (3D)</td>
</tr>
</tbody>
</table>

force) but new ones can be implemented by the user.

A successful simulation of a system requires two additional steps: the definition of an integrator that actually solves the simulation problem and the creation of reporters for the system variables. The latest allow to make available the results of the simulation as a set of storage files reporting the evolution of forces and kinematics during the analysis. About the integrator, several algorithms are available in OpenSim, with parameters about size step, accuracy and tolerance that can be changed. Among the available integrators, a good compromise between precision and computational time is the Runge-Kutta-Feldberg one [7] that was used in the simulations presented in the following.

3 MULTI-BODY SYSTEMS BENCHMARK IMPLEMENTATION

Simulating the interaction between a subject and an assistive device requires a tool proven to be highly accurate in simulating both humans and robotic systems. OpenSim is recognized to be highly reliable in the dynamic simulation of human movements but its use in the solution of mechanical problems is still limited. To provide an extensive evaluation of the software platform as a multi-body systems simulator we looked for available benchmark suites from the multi-body dynamics research community. Surprisingly, choices are still limited, being the Multi-Body System (MBS) Benchmark [9, 10] one of the most interesting proposal. Five problems compose the benchmark suite. Each problem targets a specific challenge as shown in Tab. 1. The following of this section shortly introduces the five problems. The interested reader is referred to [9, 10] for further information and to the GitHub Repository [18] for details about their implementation in OpenSim.
Table 2. N-four-bar mechanism: system properties and configuration.

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$N$</td>
<td>40</td>
</tr>
<tr>
<td>Link mass</td>
<td>1.0 kg</td>
</tr>
<tr>
<td>Link length</td>
<td>1.0 m</td>
</tr>
<tr>
<td>$\dot{B}_0 + (0)$</td>
<td>1.0 m/s</td>
</tr>
</tbody>
</table>

Figure 3. $N$-four-bar mechanism: (a) sketch and (b) OpenSim model showing only the first three windows. Reference point: $B_0$.

A01: Simple Pendulum

The first problem is a simple pendulum (Fig. 2(a)), a planar mechanism composed of a point mass linked to the ground through a rigid massless bar. The only force applied to the mechanism is the gravity. Tab. 2(c) reports system configuration while Fig. 2(b) shows a snapshot of the OpenSim implementation. This problem does not have a real challenge, and it is only proposed as a demonstration example.

A02: N-Four-Bar Mechanism

The second problem is a $N$-four-bar mechanism composed of $2N+1$ links (Fig. 3 and Tab. 2), an extension of the 2-four-bar mechanism proposed in [1]. When the mechanism reaches the horizontal position, the number of the degrees of freedom instantaneously increase from 1 to $N+1$. Direction of the gravity force is towards negative $y$. This problem is challenging because it is a common example of a mechanism which undergoes singular configurations [9].

A03: Andrew’s Mechanism

The third problem is an Andrew’s mechanism [17] (Figs. 4, 5(a)). It is a planar system composed of seven bodies interconnected through revolution joints and driven by a motor located in $O$. Tabs. 3–6 report detailed information about the mechanical structure of each body. Its simulation requires a very short time scale, thus making it challenge to simulate for solvers.
Figure 4. Andrew’s mechanism: (a) sketch and (b) OpenSim model. Reference point: F.

Figure 5. Andrew’s mechanism: (a) reference systems for the mechanism bodies and (b) initial angle values.

Table 3. Andrew’s mechanism: rod elements properties.

<table>
<thead>
<tr>
<th>Center of Mass (CoM)</th>
<th>Mass [Kg]</th>
<th>Inertia (CoM) [Kgm²]</th>
<th>Length [m]</th>
</tr>
</thead>
<tbody>
<tr>
<td>X [m]</td>
<td>Y [m]</td>
<td></td>
<td></td>
</tr>
<tr>
<td>OF 0.00092</td>
<td>0</td>
<td>0.04325</td>
<td>2.194e⁻⁶</td>
</tr>
<tr>
<td>FE -0.0115</td>
<td>0</td>
<td>0.00365</td>
<td>4.41e⁻⁷</td>
</tr>
<tr>
<td>EG 0</td>
<td>0.01421</td>
<td>0.00706</td>
<td>5.667e⁻⁷</td>
</tr>
<tr>
<td>AG 0.02308</td>
<td>0.00916</td>
<td>0.0705</td>
<td>1.169e⁻⁵</td>
</tr>
<tr>
<td>AH -0.00449</td>
<td>-0.01228</td>
<td>0.05498</td>
<td>1.912e⁻⁵</td>
</tr>
<tr>
<td>HE -0.01421</td>
<td>0</td>
<td>0.00706</td>
<td>5.667e⁻⁷</td>
</tr>
</tbody>
</table>

Table 4. Andrew’s mechanism: triangular element properties, points defined in X_{BDE}-Y_{BDE} reference system.

<table>
<thead>
<tr>
<th>Center of Mass (CoM)</th>
<th>Mass [Kg]</th>
<th>Inertia [Kgm²]</th>
<th>Point X [m]</th>
<th>Y [m]</th>
</tr>
</thead>
<tbody>
<tr>
<td>X [m]</td>
<td>Y [m]</td>
<td>Kg m²</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.01043</td>
<td>-0.01874</td>
<td>0.02373</td>
<td>B 0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5.255e⁻⁶</td>
<td>D 0.02</td>
<td>-0.018</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>E 0</td>
<td>-0.035</td>
</tr>
</tbody>
</table>
**Table 5.** Andrew’s mechanism: points in ground X-Y reference system.

<table>
<thead>
<tr>
<th>Point</th>
<th>X [m]</th>
<th>Y [m]</th>
</tr>
</thead>
<tbody>
<tr>
<td>O</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>A</td>
<td>-0.06934</td>
<td>-0.00227</td>
</tr>
<tr>
<td>B</td>
<td>0.03635</td>
<td>0.03273</td>
</tr>
<tr>
<td>C</td>
<td>0.014</td>
<td>0.072</td>
</tr>
</tbody>
</table>

**Table 6.** Andrew’s mechanism: system properties and initial configuration.

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spring coefficient</td>
<td>4530 N/m</td>
</tr>
<tr>
<td>Spring rest length</td>
<td>0.07785 m</td>
</tr>
<tr>
<td>Motor torque</td>
<td>0.033 N m^{-1}</td>
</tr>
</tbody>
</table>

**A04: Bricard’s Mechanism**

The fourth problem is the Bricard’s mechanism [3], a system composed of five rods (1.0 m length and 1.0 kg mass) and six revolute joints (Fig. 6). Gravity force is towards the negative y direction. The challenge of this problem is being an over-constrained system. Indeed, while Grubler’s formula [11] results in no degrees of freedom, the particular orientation of the revolute pairs produces a system with one degree of freedom.

**Figure 6.** Bricard’s mechanism: (a) sketch and (b) OpenSim model. Reference point: \( P_3 \).

**A05: Flyball Governor**

The last problem is also known as flyball governor (Fig. 7) and was invented by J. Watt in the 18th century. Gravity acts in the negative y direction and the system moves under its effect. In this stiff mechanical system, coupler rods are substituted by spring-damper elements, thus making challenging its simulation.

**4 EXPERIMENTAL RESULTS**

**4.1 Accuracy Evaluation**

To assess the accuracy of OpenSim APIs in implementing the problems proposed by the MBS benchmark suite, we evaluated the obtained results following the guidelines presented by the MBS authors in [9]. For each problem, the authors provided the 3-dimensional displacement of a reference point, shown in the problem sketches of the previous section. The accuracy of the coordi-
\[ g = 9.81 \text{N/kg} \]

\[ \omega = \pi \text{rad/s} \]

\[ \rho = 3000 \text{kg/m}^3 \]

\[ K = 8 \times 10^5 \text{N/m} \]

\[ C = 4 \times 10^4 \text{Ns/m} \]

\[ s = 0.5 \text{m} \]

\[ \alpha = 30^\circ \]

\[ \dot{\omega} = 2\pi \text{rad/s} \]

\[ e_j(t_i) = \frac{|y_i(t_i) - y_{i}^{ref}(t_i)|}{\max\{|y_{i}^{ref}(t_i)|, y_{threshold}^j\}} \] (1)

where \( y \) is the simulation output and \( y_{ref} \) is the reference. The threshold was introduced to avoid a singularity when the reference values approach zero. The threshold value was set to \( 10^{-5} \) for problem A03 and to \( 10^{-3} \) for the others. The choice depends on the values that the coordinates assume during the motion of the mechanism, which are lower for the A03 problem.

Eq. 2 defines the total error \( (e_{Total}) \) of the problem:

\[ e_{Total} = \sqrt{\frac{1}{m} \sum_{i=1}^{m} \frac{1}{n} \sum_{j=1}^{n} (e_j(t_i))^2} \] (2)

where \( m \) is the dimension of the displacement vector of the reference point and \( n \) the sample numbers.

### 4.2 Results

Tab. 8 reports the errors for the five problems. Overall, error values are always low. The highest value for A03 is due to the highest complexity of the motion that requires a small time step. Tab 9 reports the Root Mean Square Error (RMSE) and the Peak Error (PE) for the reference coordinates of the five problems. Once again the results demonstrate the high precision of the simulations and confirm the suitability of OpenSim APIs to implement multi-body system simulators. \( R^2 \) values
Table 8. Global error (%) of OpenSim simulations.

<table>
<thead>
<tr>
<th></th>
<th>A01</th>
<th>A02</th>
<th>A03</th>
<th>A04</th>
<th>A05</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>3.6E-3</td>
<td>9.8E-4</td>
<td>4.7E-2</td>
<td>6.4E-4</td>
<td>7.3E-5</td>
</tr>
</tbody>
</table>

Table 9. RMS and Peak errors of OpenSim simulations.

<table>
<thead>
<tr>
<th></th>
<th>A01</th>
<th>A02</th>
<th>A03</th>
<th>A04</th>
<th>A05</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMSE X [m]</td>
<td>2.8E-5</td>
<td>7.9E-5</td>
<td>7.0E-6</td>
<td>2.1E-5</td>
<td>4.0E-5</td>
</tr>
<tr>
<td>RMSE Y [m]</td>
<td>2.9E-5</td>
<td>5.4E-5</td>
<td>8.0E-6</td>
<td>2.0E-6</td>
<td></td>
</tr>
<tr>
<td>RMSE Z [m]</td>
<td></td>
<td></td>
<td></td>
<td>4.0E-6</td>
<td></td>
</tr>
<tr>
<td>Peak Error X [m]</td>
<td>5.6E-5</td>
<td>2.1E-4</td>
<td>1.6E-5</td>
<td>1.9E-4</td>
<td>1.8E-3</td>
</tr>
<tr>
<td>Peak Error Y [m]</td>
<td>5.9E-5</td>
<td>1.8E-4</td>
<td>1.9E-5</td>
<td>5.1E-6</td>
<td></td>
</tr>
<tr>
<td>Peak Error Z [m]</td>
<td></td>
<td></td>
<td></td>
<td>5.3E-5</td>
<td></td>
</tr>
</tbody>
</table>

were also computed and are always really close to the unit value ($R^2 > 0.999$) for each problem. The perfect match between simulated and reference solutions is also shown by graphs in Fig. 8.

The very good results obtained by OpenSim in simulating the problems proposed in MBS benchmark lead us to conclude that OpenSim can be used successfully to achieve accurate simulations of multi-body mechanical systems.

5 DISCUSSIONS AND FUTURE WORKS

This work presented a formal evaluation of OpenSim as mechanical multi-body system simulator. The C++ implementations of the problems using OpenSim APIs are freely available as a GitHub repository [18] and movies showing the evolution of the simulations are available at the YouTube channel of our research group [19]. We provided the source code as they are also a good starting point to model other mechanical systems in OpenSim.

Reported results shows that OpenSim already provides all the tools required to define a methodology to predict the dynamic behavior of mechanical multi-body systems. This is of fundamental importance for the development of a multi-level model that incorporates both the human neuro-musculoskeletal level [13, 15] and the mechatronic level. In this context, a work, more focused on real robotic assistive device dynamic movement simulation, has been already presented [21].

6 ACKNOWLEDGMENTS

This research has been partially supported by EU-F7 grant BioMot (project no. 611695).
**Figure 8.** Comparison between OpenSim simulation outputs (dashed lines) and MBS benchmark reference (gray lines or dots) for the reference point. Note: OpenSim output perfectly match the reference solution therefore lines are often overlapped.
REFERENCES


Multi-Body Dynamics Benchmarks for Frictionless Elastic Collisions

Reinier W. Bos*, Jeroen Wikkerink#

* Faculty of Mechanical, Maritime and Materials Engineering
  Delft University of Technology
  Mekelweg 2, 2628CD Delft, The Netherlands
  R.W.Bos@student.tudelft.nl
# Maritime Research Institute Netherlands
  Haagsteeg 2, 6700AA Wageningen, The Netherlands
  [R.Bos, J.Wikkerink]@marin.nl

ABSTRACT
Multi-body simulations are widely used in a variety of research fields. In the maritime industry, for instance to predict the behaviour of a ship during side-by-side mooring, typically with few bodies. Recently an increased need from the industry is to expand those simulations to predict the behaviour or provide training for a floating structure in ice. Benchmarks are required to quantify the performance of the simulations. Benchmarks for bilateral constraints and friction exist, but for contact problems (unilateral constraints) they are absent, to the best of our knowledge. Therefore five new possible benchmarks are given, which can be used to test the accuracy of elastic friction-less collision response. Conservation of momentum is used to derive the analytical solution, which is usable for resolving multiple simultaneous collisions. A practical example of the use of these benchmarks is given.

Keywords: Benchmark, elastic collision, multi-body.

1 INTRODUCTION
Multi-body simulations are employed for training and engineering purposes. Under these principles many real-life applications may be captured, for instance a realistic driving simulator. All simulations have in common that multiple rigid bodies are interacting with each other, by applying uni- or bilateral constraints. Unilateral constraints (or inequality constraints) act in only one direction: it is not possible to walk through a wall, but it is possible to walk away from it. Bilateral constraints (or equality constraints) act in all directions, such as a door hinge.

Systems subject to bilateral constraints can often be rewritten as a function of generalized coordinates, which uniquely describes the state of a system. In simpler cases the bilateral constraints can be eliminated by substitution. This is however not possible for unilateral constraints, which require by default an iterative procedure for which linear complementary programming is often used. An extensive description of contact dynamics and solution methods is given by Lacoursière[8].

Multi-body simulations are powered by multi-body solvers, which detect and resolve the contact between the bodies. Many different multi-body solvers are available, both commercial and open-source codes are available. These solvers can be integrated in other simulation software. At MARIN, this is done for an in-house developed framework. This framework is built to simulate the rigid-body dynamics of ships and offshore structures in waves and wind. To extend the simulation capabilities to allow for structure-ice interaction a multi-body solver was implemented. The end goal of such an implementation is an accurate simulation of an ice-going vessel with a large number of ice blocks, as shown in Fig. 1. Both the ship and the ice blocks are represented by rigid bodies.
Software development benefits from benchmarks, especially concerning unit tests in continuous integration. Benchmarks can be divided in three categories:

- **Speed**: how long does it take to reach a result?
- **Robustness**: is it possible to converge to a result?
- **Accuracy**: what is the error?

Depending on the application preference can be given to any of these points. For training purposes speed is important, since the training has to be done in real time. Engineering purposes require high accuracy of a simulation. A solver often has some input which can be tweaked to give preference to either of these points. For instance, a smaller timestep reduces the error but increases calculation time.

In order to evaluate any of these three points some ‘reference’ has to be generated. For robustness and speed it is possible to compare different solvers and see whether they are able to come up with a result and how long this takes. However, both results might be wrong. Accuracy requires a reference solution which is known to be true. This is an ideal point for a unit test, since versions of software can objectively be compared against the reference solution or benchmark.

Several benchmarks for multi-body collisions are available. These benchmarks often evaluate the speed and robustness of a contact solver, as stated by González et al.[1]. A number of quantitative benchmark cases exist for multi-body problems, which are collected in a database by Masoudi et al.[5], and another one by Gonzalez et al.[1, 2]. These cases are all dynamic, conservative and test the dynamics engine on its bilateral (equality) constraint handling. Some benchmark cases which evaluate contact behaviour, such as by Seugling and Rölin[3], are purely qualitative. To the best of our knowledge benchmarks for accuracy of contact (unilateral) problems are lacking.

In this paper a number of benchmarks is proposed for frictionless elastic collisions. To knowledge of the authors, this has not yet been done for multiple simultaneous collisions. First, benchmark candidates are developed. Then, for comparison and verification a suitable analytical solution is derived. Finally the practical use to verify the contact solver in an in-house developed framework is shown.
2 BENCHMARK CANDIDATES

By definition, a benchmark is a comparison, in this case of a numerical simulation with an analytical result. This means that whatever kind of benchmark is chosen, an analytical result should be calculable. It should also be possible to simulate the same situation in the solver that is investigated. Preferably the benchmark stands on its own: it should not include any physics unrelated to those investigated. In this case it means that the benchmark only contains a collision problem and effects such as friction or inelastic (non-conservative) behaviour are excluded. A benchmark should be as simple as possible to make sure it can be practically used, to single out a problem.

The benchmarks are developed for software which should predict ship-ice interaction. Ship-ice interaction is characterized by large mass ratios: the ship is often much larger and heavier than the ice. The contact problem involves both friction and collision, of large amounts of bodies. The benchmarks should be able to predict behaviour of large amounts of bodies, with high mass ratios.

Consider a multi-body simulation where each body $B_i$ has a state described by position $\vec{x}$ and rotation $\vec{\theta}$. Both linear and angular momentum are conserved throughout the simulation for all bodies[7]:

$$\dot{m}\vec{x}^- + \sum\int_{t^-}^{t^+} \vec{F} dt = m\dot{\vec{x}}^+$$

(1)

$$I\dot{\vec{\theta}}^- + \sum\int_{t^-}^{t^+} \vec{M} dt = I\dot{\vec{\theta}}^+$$

(2)

with $m$ denoting the body mass, $I$ the inertia matrix for rotation, $\vec{F}$ an external force and $\vec{M}$ an external moment. The dot $\dot{}$ denotes a time derivative, so that $\dot{\vec{x}}$ is the velocity and $\dot{\vec{\theta}}$ is the rotation velocity (shorthand spin). The time indications are $t^-$ for the interval prior to collision, and $t^+$ for the time after collision. These superscripts are also used for other variables.

Only collision forces are taken into account, which are equal and opposite for both bodies. The momentum equations are not sufficient to uniquely determine the collision response, because the reaction forces are unknown. An additional condition for elastic collisions is that no energy is lost due to the collision. Hibbeler[7] uses a coefficient of restitution, to mimic the loss of energy in a collision. Here, such a coefficient of restitution is set to unity to conserve kinetic energy over time:

$$\frac{\partial E_k}{\partial t} = 0, \quad E_k = \frac{1}{2}\sum_{\text{bodies}} (m|\dot{\vec{x}}|^2 + \dot{\vec{\theta}}^T I \dot{\vec{\theta}})$$

(3)

For sake of simplicity the benchmarks are built around two blocks, a large (cube of $10 \text{ m}$) and a small one (cube of $1 \text{ m}$). Regardless of the benchmarks, he small blocks move towards the large one. This prevents interaction after the first collision. At the start of the simulation the large block is stationary and the smaller blocks move towards the large block with a constant speed of $1 \text{ m/s}$ and perpendicular to the normal of the closest surface of the big block. The small blocks are placed $1 \text{ m}$ away from the big block, which means that they will impact $1 \text{ s}$ after the start of the simulation. The small blocks also impact not directly on the corner, but $1 \text{ m}$ away from it, to prevent collisions of the small blocks among themselves.

Five benchmarks are set up as shown in Figure 2 and will now be treated from left to right. The first benchmark only has linear motion and no rotation, as the impact of the small blocks occur

---

1 Naturally, the benchmarks do not depend on any units, and all consistent sets of mass, length and time can be used. Here kg, m and s are used, mostly to show the units.
simultaneously and symmetrically. The second benchmark has rotation and translation of the large block and only translation of the small block. Benchmark three has only linear motion of all the blocks, since the three small blocks impact symmetrically. In the fourth and fifth benchmark the large block remains stationary, and the small blocks should bounce back with equal velocity.

These five benchmarks cover an entire region of collisions. Evaluation of these benchmarks could be done in a straightforward way, namely by checking if the linear and angular momentum are constant over time. This is however not sufficient. Two blocks can collide and bounce back with a much higher velocity. As long as the velocity magnitudes are equal and direction opposite, the conservation of momentum is satisfied. The introduction of an energy balance alleviates this problem: the kinetic energy of each body is evaluated over time and the sum should be constant. However, this still does not make the response direction unique for collisions which are not head-on, which will be dealt with in the next section.

A note on friction: Within this project, friction was also evaluated. This was done in the same manner Sukhurukov and Loset[4] investigated real ice-ice sliding. In this case, the benchmark was performed by fixing a large block and letting a small block slide over it, under a constant normal force and linear increasing in-plane force. The small block will start to move after the in-plane force is larger than the static friction force. The time at which this happens denotes the force.

3 ANALYTICAL SOLUTION
In this section a derivation of the elastic collision response of multiple rigid bodies is presented. This derivation follows roughly the same line as Ermolin and Kazakov[6]. First a single collision
Figure 3: Overview of the kinematics of body $\mathcal{B}_a$, right before collision with body $\mathcal{B}_b$.

between two bodies is investigated. Then the multiple collisions are investigated.

3.1 Single collision between two bodies

As stated previously, the analytical solution should comply with Eqs. 1, 2 and . Furthermore, it should be energy conservative. Friction has already been excluded from the problem, so the only external force acting on a body is due to collision with another body. Hereafter, a collision between two bodies will be denoted as a collision between body $\mathcal{B}_a$ and $\mathcal{B}_b$, with the velocity and rotation velocity likewise complemented with a subscript and calculated from the centre of mass. This is shown in Fig. 3.

Here it is assumed that the contact between the two bodies happens at a mere point, instead of an area or line. After all, a force acting on a line or area can be replaced by a resultant force. Velocity $\vec{v}_a$ is the velocity of body $\mathcal{B}_a$ at the point of contact, during the collision. Only the elastic collisions are examined, thus no friction is present. By definition, the friction acts normal to the contact point and at the contact point the bodies have equal but opposite normals $\vec{n}_a = -\vec{n}_b$. This allows for a decomposition of the velocities at the contact point in a normal and tangential direction:

$$\vec{v}_{a,n} = \vec{v}_a \cdot \vec{n}_a \quad (4)$$
$$\vec{v}_{a,t} = \vec{v}_a - \vec{v}_{a,n} \quad (5)$$

and the contact force can be decomposed similarly in $\vec{F}_{a,n}$ and $\vec{F}_{a,t}$, to make the total contact force $\vec{F}_a = \vec{F}_{a,n} + \vec{F}_{a,t}$. Because there is no friction, the tangential component of the contact force is zero, so:

$$\vec{F}_a = \vec{F}_{a,n} = F_a \vec{n}_a \quad (6)$$

and, as stated previously, the force acting on the other body is equal in magnitude and opposite in direction, so $F_a$ is written as $F_{ab}$, the contact force between bodies $\mathcal{B}_a$ and $\mathcal{B}_b$. The state vector $\vec{u}$ is used to describe the state of a body, containing the position and rotation:

$$\vec{u} = \begin{bmatrix} \vec{x} \\ \theta \end{bmatrix} \quad (7)$$
which makes gives the following inertia matrix:

\[
M = \begin{bmatrix}
mI_{3 \times 3} & 0 \\
0 & I
\end{bmatrix}
\] (8)

where \(I_{3 \times 3}\) is the 3 by 3 identity matrix, \(m\) is the mass and \(I\) the inertia, as defined before. The equation for the moments of force can also be simplified, using the assumption of a point of contact. It is assumed that no moment of force is transmitted through the contact point, only a force. Then, the moment of force acting on the body is defined as:

\[
\vec{M} = \vec{r} \times \vec{F}
\] (9)

with \(\vec{r}\) as vector pointing from the centre of mass of the body to the point of application of the force. The cross product can be rewritten in matrix form as:

\[
\vec{M} = \Omega \vec{F}, \quad \Omega = \begin{bmatrix}
0 & -r_3 & r_2 \\
r_3 & 0 & -r_1 \\
-r_2 & r_1 & 0
\end{bmatrix}
\] (10)

and therefore the momentum equations are rewritten to:

\[
M \ddot{\vec{u}} + \sum \int_{t^-}^{t^+} C \vec{F} dt = M \ddot{\vec{u}}^+
\] (11)

\[
C = \begin{bmatrix}
I_{3 \times 3} \\
\Omega
\end{bmatrix}
\] (12)

where Eq. 6 can be inserted to have only the force magnitude as unknown.

The collision of the two bodies has three stages: 1) the bodies are moving towards each other, 2) the collision a collision and 3) the bodies move away from each other. From here on the point in time at which the collision takes place is denoted as \(t^*\). Here a point in time is explicitly used for convenience in notation and is valid because all bodies are rigid. Hibbeler[7] uses an undefined period in time, but in his case the bodies are not necessarily rigid, which means that the collision can take a longer time than an infinitesimal duration. It is also possible to write the following:

\[
\int_{t^-}^{t^*} \vec{F} dt = \int_{t^-}^{t^*} \vec{F} dt + \int_{t^*}^{t^+} \vec{F} dt
\] (13)

and let for instance \(t^- = t^* - \varepsilon\), then take limit of \(\varepsilon\) to 0. Note that because the collision happens at a point in time, the time derivatives of the normal are zero during contact.

The first integral on the right hand side is considered, where the following condition describes the collision, or state at \(t = t^*\):

\[
\hat{n}_a^T (C_a^T \hat{u}_a - C_b^T \hat{u}_b) = 0
\] (14)

with \(C_a\) as defined in Eq. 12 for the collision point relative to the centre of mass of body \(B_a\). This equations means that the velocity in normal direction of the contact point should be the same for both bodies. This is a valid since the velocity at a point due to rotations is:

\[
\hat{\theta} \times \vec{r} = -\vec{r} \times \hat{\theta} = \Omega \vec{r} \hat{\theta}
\] (15)

Equation 6 is substituted into Eq. 11, which is then substituted in Eq. 15. In this equation the reaction force \(F_{a,b}\) is the only unknown, yielding the following equation:

\[
F_a = \frac{\hat{n}_a^T (C_a^T \hat{u}_a - C_b^T \hat{u}_b)}{\hat{n}_a^T (C_a^T M_a^{-1} C_a + C_b^T M_b^{-1} C_b) \hat{n}_{ab}}
\] (16)
The collision force is calculated from the point of \( B_a \). If the force was calculated with the bodies swapped, then it would be negative due to the minus sign in the numerator, which is intuitively right because the forces are equal and opposite. In order to stop the bodies from penetrating the reaction force has to be applied once. The reaction force calculated in Eq. 16 yield the force required to enforce Eq. 3.2. To obtain the state after collision and be completely conservative the reaction force is applied a second time, yielding:

\[-2M_a^{-1}C_a^T \hat{n}_a F_a + \hat{u}_a = \hat{u}_a^+\]  

(17)

### 3.2 Multiple simultaneous collisions

The approach for a single collision between two bodies is now expanded to multiple collisions. All underlying assumptions are the same: the collision happens at a point in space and time, the collision is conservative and occurs without friction. Instead of a single external force on a body, multiple external forces act: one for each collision the body is involved in. The collisions will be numbered by index \( i \). The body designations \( a \) and \( b \) are evaluated for each collision.

For a body, the momentum equation can also be written as follows:

\[
\sum_i C_{a(i)}^{(i)} \hat{n}_{a(i)} F_{a(i)} = M_a (\hat{u}_a - \hat{u}_a^+) \]  

(18)

with \( i \) as collision force index, one for each collision. The summation is required, since multiple collisions can act on one body. Naturally, \( F_{a(i)} \) equals zero when the body is not involved in that collision.

The non-penetration condition (Eq. 3.2) can again be applied, and substituting Eq. 18 yields:

\[
\hat{n}_{(j)^T} \left( C_{a(j)}^{(j)} \sum_i M_a^{-1} C_{a(i)}^{(i)} T \hat{n}_{a(i)} F_{a(i)} - C_{b(j)} \sum_i M_b^{-1} C_{b(i)}^{(i)} T \hat{n}_{b(i)} F_{b(i)} \right) = \hat{n}_{(j)^T} (C_a \hat{u}_a - C_b \hat{u}_b) \]  

(19)

where the subscript \( a, b \) denotes the two bodies involved in collision \( j \). Note each collision yields one non-penetration condition, and thus one instance of Eq. 19. This complex equation can be rewritten as a system of linear equations, since the only unknowns are the collision reaction forces and the equations are linear in these forces. The following can be obtained:

\[
\sum_i A_{ji} F_i = b_j \]  

(20)

\[
A_{ji} = \hat{n}_{(j)^T} \left( C_{a(j)}^{(j)} T \hat{u}_a + C_{b(j)}^{(j)} T \hat{u}_b \right) \]  

(21)

\[
b_j = \hat{n}_{(j)^T} \left( C_{a(j)}^{(j)} T \hat{u}_a - C_{b(j)}^{(j)} T \hat{u}_b \right) \]  

(22)

with \( j \) as collision condition index (Eq. 3.2), one for each collision. The resulting forces can be substituted in Eq. 17, which now turns into a summation over all collisions. Note that \( A_{ji} \) has zero values if certain collisions are not connected via bodies.

### 4 USE OF THE BENCHMARK

This section describes the practical use of the aforementioned benchmarks cases. Here, MARIN’s in-house developed framework XMF was integrated with the Vortex Physics Engine (©CMLabs). Both software packages are widely used for various fields of study and known for their quality. Difference between results that were discovered are mostly due to difficulties in coupling two simulator worlds, which is often not straightforward. The results shown hereafter illustrate the practical use of the benchmark cases. They do not show the behaviour of the final software. Some of the differences between the benchmarks and the implementation of the physics engine in the software framework require further investigation.
4.1 Implementation and error measures

Five benchmark cases have been programmed. In the first set-up the iterative solver was used with an older version of Vortex with a time step of $dt = 0.1\, s$, a coefficient of restitution of 1, compliance of $10^{-9}$, contact tolerance of $10^{-3}$ and friction turned off. In the second set-up a more recent version of Vortex is used with the linear complementarity solver; all other parameters are the same. The tests can be performed for a range of mass ratios between the large and small bodies, depending on the use of the simulation. Large mass ratios are simulated because they are relevant for ship-ice interaction.

In this case a systematic series will be tested: mass ratios $(m_{\text{large}}/m_{\text{small}})$ from $10^{-2}$ to $10^4$ are taken and for each case three ‘performance monitors’ are evaluated:

1. Error with respect to analytical solution: absolute error in velocity and rotation velocity
2. Conservation of momentum: absolute error
3. Conservation of energy: relative error and root mean square over time

For the first performance monitor the absolute error is used, as the value of velocity and rotation velocity should be zero for some cases. This would hence lead to an incalculable error if a relative error was used. For the same reason the absolute error is used as third performance monitor. The second performance monitor uses conservation of energy. As the benchmarks are set up in a way that only allows interaction of the bodies in one collision, this is simply evaluated by taking the total kinetic energy before and after the collision, scaled with the initial energy. If the difference is zero, the relative error is zero. The root mean square difference with respect to the mean kinetic energy is used to quantify any errors over time, so in the correct case this should be zero.

4.2 Results

Figure 4a shows the result of the systematic testing for the first iteration. Analysing these results is not trivial, as there are 90 tests shown, with 2 parameters per test describing the result. Overall, large deviations of the numerical solution to the benchmark are observed. At least for the mass ratio of 1 (large mass over small mass = 1) a good result is expected. Investigation of these results showed that a small adjustment of the integration resulted in dramatic improvements of the results. The improved results are shown in Fig. 4. The following can be said of this version:

- Figure 4b shows that large absolute errors are made for the velocity magnitude, even with mass ratios of order unity.
- Figure 4b shows that the case 2 has large errors for the rotation velocity magnitude, but these errors decrease with increasing mass ratio.

The numerical results are not yet in correspondence with the benchmark prediction. Therefore the other error norms are also considered.

- Figure 5a shows that the difference between initial and final linear momentum is extremely small for all mass ratios.
- Figure 5a shows that angular momentum is reasonably well conserved. Case 2 is the only case which should induce rotations and shows the highest error, which is due to usage of the absolute error. Case 3 shows for low mass ratios the comparable absolute erro to case 2 and is therefore worse.
Figure 5b shows that large errors are made for kinetic energy, in cases where the final movement of the big block should be zero. All errors seem to converge, but increase again for large mass ratios.

Figure 5b shows that the root mean square is non-zero, which means that the kinetic energy is not constant during the simulation. This can be expected based on the previous point.

In general, no conclusion can be drawn on the behaviour of the simulations. The results for mass ratios \( \geq 1 \) seem to be quite accurate.

The last statement of the list is quite surprising. Physics engines are often designed for mass ratios close to unity, and these results are a pleasant surprise for the application to ship-ice interaction. Here the benchmarks shows that this implementation can be used outside the mass range for which it is intended.

Another note on friction: The friction cases were also evaluated two times. The first time a lot of jittering and falling-through occurred. It was found that this was due to the handling of constraints. After improving this, the results matched exactly what was expected.

5 CONCLUSIONS

To have reliable results of multi-body dynamics simulations five benchmarks for elastic collisions are proposed. These benchmarks involve linear motion, rotations and multiple simultaneous collisions between multiple bodies. An analytical method is developed to predict the result of the benchmark problems. The benchmark has been used successfully to improve the integration of two software packages. In the future these benchmarks will be used to further develop the coupling between in-house software and an external physics engine.

REFERENCES


(a) Absolute error of end velocity with respect to analytical solution, intermediate result.

(b) Absolute error of end velocity with respect to analytical solution, final result.

Figure 4: Systematic testing of the cases using mass ratios of $10^{-2}$ to $10^4$. The top figure corresponds to an older configuration, which was improved with help of the presented benchmarks to obtain the bottom result.
Figure 5: Systematic testing of the cases using mass ratios of $10^{-2}$ to $10^4$ using the improved version, to check for other errors.
Testing the efficiency and accuracy of multibody-based state observers

Emilio Sanjurjo *, José L. Blanco #, José L. Torres #, Miguel A. Naya *

* Laboratorio de Ingeniería Mecánica
University of La Coruña
Mendizábal s/n, 15403 Ferrol, Spain
[emilio.sanjurjo, minaya]@udc.es

# Escuela Politécnica Superior
University of Almería
La Cañada de San Urbano, 04120 Almería, Spain
[jiblanco, jltmoreno]@ual.es

ABSTRACT

Multibody simulations are already used in many industries to speed up the development of new products. However, improvements in multibody formulations and the continuous increase of computational power open new fields of applications for the multibody simulations, such as using them as a basis for developing state observers. This work introduces novel state observers developed by combining a multibody model with some probabilistic estimators from the family of Kalman filters. Together with other multibody-based state observers already proposed in the literature, have been benchmarked by applying them to two mechanisms: a planar four-bar linkage, and a five-bar linkage. The accuracy of the estimations as well as the computational costs are examined under several scenarios: using encoders or gyroscopes as sensors, with different sampling rates for the sensors, and assuming different levels of modeling errors. The results obtained in this work can be employed to select a suitable state observer for a new application. All methods have been implemented as a reusable MATLAB toolkit which has been released as Open Source in https://github.com/MEDS/mbde-matlab.

Keywords: Multibody simulation, State observers, Kalman filtering, Benchmark.

1 INTRODUCTION

Multibody simulations (MBS) are a common practice in the industry to speed up the development of new products. Since many years ago, this tool was employed offline to predict the behavior of new concepts, or extreme situations which are difficult or expensive to attain with a prototype. After that, with the development of new multibody formulations and the increase of computational power, real-time multibody simulations became a reality, allowing people and machines to interact with the multibody models in simulators and virtual test benches. Nowadays, with the advent of low cost, low power consumption computers, a multibody simulation could be run in vehicles or robots as part of their control algorithms, providing information about immeasurable magnitudes. The problem with this approach is that, in general, a multibody model can be very accurate in the short term if the forces are accurately known, but it will diverge over the time.

For this reason, multibody models should be corrected with information from the actual mechanisms in order to be used as reliable state observers. One way to reach this aim is employing information fusion techniques to combine information about multibody models with data provided by real sensors installed on the machine. A prominent example of information fusion technique is the Kalman filter (KF) which, however, presents some problems regarding its direct application to this problem. Firstly, the KF was originally formulated for first order, linear and unconstrained models, whereas multibody models are, in general, second order, highly nonlinear, and constrained systems. Moreover, the algorithm should be efficient in order to be run in real time. Because of these reasons, this is an open field of research. In a previous work [1], some existing methods [2, 3], and several new formulations were tested with different modeling errors and with different levels of sensor’s noise.

In this work, two new MBS-based formulations were developed: the errorEKF, an error-state Kalman filter, and the projectionEKF, a method in dependent coordinates in which a projection technique is employed to fulfill the constraints of the multibody model, based on the methods
proposed in [4]. Their accuracy and performance was compared against the best methods presented in [1]: the CEKF and the UKF methods for their accuracy, and the DEKF and the DIEKFpm methods for their performance. The tests performed comprise different kind of sensors, including gyroscopes and encoders, different levels of modeling errors, and different sampling rates for the sensors, everything applied to two planar mechanism: a four-bar, and a five-bar linkages.

To provide a reference level for attainable accuracy, to which all these results can be compared, a batch estimator is applied to the whole history of sensor measurements, based on the non-linear optimization of the graphical model proposed in [5].

2 METHODOLOGY

At this moment there is a huge amount of different possibilities regarding the design of a state observer for a multibody model. For this reason, the only feasible approach to perform a large amount of tests is by means of simulations. For every set of tests, a multibody model is run playing the role of a real mechanism. This model provides the ground truth, and also is employed to build the signals from the sensors. To do this, perfect sensors are modeled, and then random noise is added to their measurements. A second multibody is built, but modifying some of the properties, to simulate modeling error. Usually, the geometry and the mass properties of any machine can be known with a great accuracy. However, the level of accuracy in force models is usually not so good. For this reason, the parameter which is intentionally modified is the acceleration of gravity. Finally, the state observer is built using the latter multibody model (the imperfect one), and corrected with the information provided by the sensors built from the simulation of the first multibody model.

The mechanisms selected for this benchmark are a four-bar and a five-bar linkages (see fig. 1). The four bar mechanism was selected because is the simplest closed-loop mechanism. The five bar mechanism is the next step in complexity to check that the algorithms can be generalized to any number of degrees of freedom. Gravity is the only actuating force.

Every test was run 20 times, and the results were averaged. The different tests performed comprise the two mechanisms, with tho different levels of error in the magnitude of the gravity, 4 different sampling rates for the sensors, and two different sets of sensors for every mechanism.

3 DESIGN OF THE OBSERVERS

Next we introduce the different filtering algorithms that are benchmarked in this work. The reader can refer to table 1 as a summary of the notation employed in this work.
Table 1. Notation summary.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( n )</td>
<td>Number of dependent coordinates</td>
</tr>
<tr>
<td>( m )</td>
<td>Number of constraints</td>
</tr>
<tr>
<td>( g )</td>
<td>Number of degrees of freedom</td>
</tr>
<tr>
<td>( z )</td>
<td>Vector of independent coordinates</td>
</tr>
<tr>
<td>( q = q(z) )</td>
<td>Vector of dependent coordinates</td>
</tr>
<tr>
<td>( \Phi(q) = 0 )</td>
<td>Constraint equations</td>
</tr>
<tr>
<td>( \Phi_q, \Phi_x )</td>
<td>Jacobian of ( \Phi ) with respect to ( \dot{q}, \dot{x} )</td>
</tr>
<tr>
<td>( M )</td>
<td>Mass matrix</td>
</tr>
<tr>
<td>( Q )</td>
<td>Vector of generalized forces</td>
</tr>
<tr>
<td>( x, \dot{x} )</td>
<td>Real value and estimation of the filter state vector</td>
</tr>
<tr>
<td>( \hat{x}<em>k, \hat{x}</em>{k+1} )</td>
<td>Estimation mean at time step ( k ), before and after the update stage</td>
</tr>
<tr>
<td>( P_k, P_{k+1} )</td>
<td>Estimation covariance at time step ( k ), before and after the update stage</td>
</tr>
<tr>
<td>( f(\cdot), f_x, f_q )</td>
<td>Transition model and its Jacobians w.r.t. ( \hat{x} ) and ( \dot{q} )</td>
</tr>
<tr>
<td>( h(\cdot), h_x, h_q )</td>
<td>Observation (sensor) model and its Jacobians w.r.t. ( \hat{x} ) and ( \dot{q} )</td>
</tr>
<tr>
<td>( o_k )</td>
<td>Sensor measurements at time step ( k )</td>
</tr>
<tr>
<td>( \Sigma_P )</td>
<td>Covariance matrix of system transition (&quot;plant&quot;) noise</td>
</tr>
<tr>
<td>( \Sigma_S )</td>
<td>Covariance matrix of sensors noise</td>
</tr>
<tr>
<td>( K )</td>
<td>Kalman gain matrix</td>
</tr>
<tr>
<td>( I_N )</td>
<td>The ( N \times N ) unit matrix</td>
</tr>
</tbody>
</table>

3.1 Independent coordinates methods

The Kalman filter assumes that its states are independent. For this reason, the more natural way to combine a multibody model and a Kalman filter is by using independent coordinates.

3.1.1 Continuous extended Kalman filter (CEKF)

This formulation was already described in some previous works [2] but it will be reproduced here for the convenience of readers. The main idea under this formulation is to adapt the multibody equations in order to fit the Kalman filter structure. In its most basic form, the dynamics of a multibody system is described by the constrained Lagrangian equations:

\[
\begin{align*}
\mathbf{M}\ddot{\mathbf{q}} + \mathbf{\Phi}_q^\top \mathbf{\lambda} &= \mathbf{Q} \\
\mathbf{\Phi} &= 0
\end{align*}
\]

(1)

As the multibody equations are expressed in the form of continuous-time differential equations, it seems natural to adopt the continuous-time version of the Kalman filter. The multibody formalism employed is the R-matrix formulation [6]. The main idea behind this formulation is to obtain an ODE with dimension \( g \) equal to the number of degrees of freedom, starting with the identity \( \ddot{\mathbf{q}} = \mathbf{R}\dot{\mathbf{z}} \), which relates dependent and independent velocities. Accelerations can be then expressed as follows:

\[
\ddot{\mathbf{q}} = \mathbf{R}\dot{\mathbf{z}} + \mathbf{\dot{R}}\mathbf{z}
\]

(2)

Going back to Eq. (1), premultiplying by the transpose of \( \mathbf{R} \), and having in mind that \( \mathbf{\Phi}_q\mathbf{R} = 0 \),

\[
\ddot{\mathbf{z}} = \left( \mathbf{R}^\top \mathbf{MR} \right)^{-1} \left[ \mathbf{R}^\top \left( \mathbf{Q} - \mathbf{MR}\dot{\mathbf{z}} \right) \right] = \mathbf{M}^{-1}\mathbf{Q}
\]

(3)

If now the filter state is defined as the vector \( \mathbf{x}^\top = \{ \mathbf{z}^\top, \dot{\mathbf{z}}^\top \} \), it turns out that:

\[
\begin{align*}
\begin{bmatrix} \ddot{\mathbf{z}} \\ \dot{\mathbf{z}} \end{bmatrix} &= \begin{bmatrix} \mathbf{M}^{-1}\mathbf{Q} \\ \mathbf{M}^{-1}\mathbf{Q} \end{bmatrix} \\
\mathbf{\dot{x}} &= \mathbf{f}(\mathbf{x})
\end{align*}
\]

(4)
These equations perfectly fit the continuous extended Kalman filter equation, so they can be straightforwardly applied. In particular, the state-space transition matrix is obtained as the linearization:

\[
A = \frac{\partial f}{\partial x} = \begin{bmatrix}
0 & I \\
\frac{\partial (M^{-1}Q)}{\partial z} & \frac{\partial (M^{-1}\dot{Q})}{\partial z}
\end{bmatrix}
\] (5)

which can be approximated by:

\[
A \simeq \begin{bmatrix}
0 & I \\
A_{21} & A_{22}
\end{bmatrix}
\]

\[
A_{21} = -\hat{M}^{-1}\hat{R}^T(\hat{K}\hat{R} + 2\hat{R}\hat{q}\dot{\hat{z}})
\]

\[
A_{22} = -\hat{M}^{-1}\hat{R}^T(\hat{C}\hat{R} + M\dot{\hat{R}})
\]

where \(\hat{K}\) and \(\hat{C}\) are the stiffness and damping matrices, respectively. In this case the size of the problem is \(2g\). Next, we introduce the CEKF correction stage [7], which fuses the sensor information into the filter, leading to:

\[
\dot{\hat{z}} - \hat{z} + Kz(y - o) = 0
\] (7a)

\[
\hat{M}\ddot{\hat{z}} - \hat{Q} + \hat{M}K\dot{z}(y - o) = 0
\] (7b)

In order to numerically integrate the result of the filter, the implicit single-step trapezoidal rule has been selected as integrator:

\[
\hat{z}_{n+1} = \frac{2}{\Delta t} \hat{z}_{n+1} - \left( \frac{2}{\Delta t} \hat{z}_n + \hat{z}_n \right)
\] (8a)

\[
\dot{\hat{z}}_{n+1} = \frac{2}{\Delta t} \dot{\hat{z}}_{n+1} - \left( \frac{2}{\Delta t} \dot{\hat{z}}_n + \dot{\hat{z}}_n \right)
\] (8b)

Combining Eq. (7) and Eq. (8) leads to the following nonlinear system,

\[
\begin{cases}
\begin{align*}
g_1(\hat{x}_{n+1}) &= 0 \\
g_2(\hat{x}_{n+1}) &= 0
\end{align*}
\Rightarrow g(\hat{x}_{n+1}) = 0
\end{cases}
\] (9)

This system can be iteratively solved, e.g. by means of the Newton-Raphson method, employing the following approximate Jacobian matrix:

\[
\frac{\partial g}{\partial x} = \begin{bmatrix}
\frac{2}{\Delta t}I & -I \\
R^T\hat{K}R & R^T(\hat{C}\hat{R} + M\dot{\hat{R}}) + \frac{2}{\Delta t}M
\end{bmatrix}
\]

\[
\begin{bmatrix}
K^zh & K^zh \\
MK^zh & MK^zh
\end{bmatrix}
\] (10)

where the pair \(h_z\) and \(h_{\dot{z}}\) and the pair \(K^z\) and \(K^\dot{z}\) are the position and velocity parts of the sensor Jacobian matrix and the Kalman gain matrix, respectively.

### 3.1.2 Discrete extended Kalman filter (DEKF)

This is the discrete-time version of CEKF described above. A key difference between CEKF and the rest of estimators described from now on, which work in discrete time steps, is that the filter formulation clearly consists of two separated stages: state transition (also called prediction) and state update. The former relies on the transition model of the system (dynamical equations) while the latter includes the information from sensors, or observations – this is in contrast to CEKF where both stages are seamlessly fused together.

Each stage comprises differentiated equations for updating the state vector and the covariance matrix. Starting with the prediction stage, the EKF equations in their most generic form are:

\[
\hat{x}_k^- = f(\hat{x}_{k-1}^-)
\] (11a)

\[
P_k^- = f_{xk-1}P_{k-1}f_{xk-1}^T + \Sigma_k^p
\] (11b)

where the pair \(h_z\) and \(h_{\dot{z}}\) and the pair \(K^z\) and \(K^\dot{z}\) are the position and velocity parts of the sensor Jacobian matrix and the Kalman gain matrix, respectively.
where $f(\cdot)$ stands for the transition model of the system. By considering now the state vector of a MBS estimator in independent coordinates, $\hat{x} = \{\hat{z}^T, \hat{\dot{z}}^T\}$, and assuming the usage of the Euler method for numerical integration with time step $\Delta t$, we can put the integrator in a form that fits that required by the EKF transition function $f(\cdot)$:

$$\hat{x}_k^- = f(\hat{x}_{k-1}^+) \rightarrow \begin{bmatrix} \hat{z}_k \\ \hat{\dot{z}}_k \end{bmatrix} = \begin{bmatrix} \hat{z}_{k-1} + \Delta t \hat{\dot{z}}_{k-1} \\ \hat{\dot{z}}_{k-1} + \Delta t \dot{z}_{k-1} \end{bmatrix} \quad (12)$$

Here, the only unknown term is the acceleration vector $\ddot{z}_{k-1}$ for the previous time step, which must be computed by solving the multibody equations of motions as in Eq. (3). Thus, it follows that the transition model Jacobian $f_k$ has a fairly simple structure:

$$f_k = \frac{\partial f}{\partial \hat{x}} = \begin{bmatrix} \frac{\partial}{\partial \hat{z}} \hat{z} + \Delta t \hat{\dot{z}} \\ \frac{\partial}{\partial \hat{\dot{z}}} \hat{\dot{z}} \end{bmatrix} = \begin{bmatrix} I_g & \Delta \dot{t} I_g \\ 0_{g \times g} & I_g \end{bmatrix} \quad (13)$$

Regarding the $\Sigma^p_{k-1}$ covariance matrix appearing in Eq. (11), it stands for the additional uncertainty of the new state $\hat{x}_k$, physically attributable to unmodeled forces and errors in the parameterization of the mechanism (e.g. lengths of bars, inertia values, etc.). Assuming independent and identically distributed (iid) Gaussian noise for each independent coordinate, its structure becomes:

$$\Sigma^p_{k-1} = \begin{bmatrix} \sigma^2_\alpha I_g & 0_{g \times g} \\ 0_{g \times g} & \sigma^2_\beta I_g \end{bmatrix} \quad (14)$$

with the parameters $\sigma_\alpha$ and $\sigma_\beta$ specifying the standard deviations of the assumed noise in position and velocities, respectively.

The second stage of the DEKF method, the update, incorporates the sensor readings to improve the estimate:

$$\begin{align*}
\hat{y}_k &= o_k - h(\hat{x}_k^-) \quad (15a) \\
S_k &= h_k^T P_k^- h_k + \Sigma^S_k \quad (15b) \\
K_k &= P_k^- h_k^T S_k^{-1} \quad (15c) \\
\hat{x}_k^+ &= \hat{x}_k^- + K_k \hat{y}_k \quad (15d) \\
P_k^+ &= (I_g - K_k h_k^T) P_k^- \quad (15e)
\end{align*}$$

where $h(\cdot)$ stands for the observation model of the system, such that $\hat{y}_k$ in Eq. (15) is clearly the error or mismatch (often called innovation) between the expected sensor readings and their actual values ($o_k$). The covariance matrix $S_k$ in Eq. (15b), or innovation covariance, represents the uncertainty in the system state projected via the sensor function ($h_k^T P_k^- h_k$) plus an additional additive Gaussian noise originated in the sensor itself ($\Sigma^S_k$). Small values of $S_k$ mean that the observation introduces useful information to constrain the estimation of the system state. By evaluating the temporary term known as Kalman gain ($K_k$) we can update the estimate mean and covariance, in Eq. (15d) and Eq. (15e), respectively. These values are then used as the input to the next iteration of this iterative filter in the next time step.

### 3.1.3 Unscented Kalman filter (UKF)

The Unscented Kalman Filter (UKF) [8] is an evolution of the family of Kalman filters that is better suited to cope with strong nonlinearities in the transition and observation models. Comprising the same prediction and update stages than DEKF, the differentiating feature of UKF is the avoidance of the first order Taylor approximation in the propagation of Gaussian random variables through the transition and observation functions. Instead, a set of samples are deterministically-chosen from the Gaussian distributions, transformed via the corresponding function, then those samples in the transformed space converted back into a parametric distribution, i.e. they are used to compute
the mean and covariance of the corresponding Gaussian. As shown in [8], this approach captures the correct posterior mean and covariance up to the third order of a Taylor series expansion, in contrast to the first order of DEKF and most other methods. In turn, its computational cost is in general higher than simpler methods.

For the present benchmark, the state vector of UKF comprises the independent coordinates and their velocities, that is, $\hat{x}^\top = \{\hat{x}^\top, \hat{\dot{x}}^\top\}$. As mentioned above, each filter iteration comprises the same two steps than DEKF, so only the differences will be highlighted here. Denoting the dimensionality of the state space $|\hat{x}|$ as $L$, a total of $2L + 1$ deterministic samples (or sigma points) $\chi_i$ with $i = 0, ..., 2L$ are generated from the mean $\hat{x}_{k-1}$ and covariance $P_{k-1}$, each with a different weight $W_i$. Then, the samples are transformed with a forward Euler transition function identical to that of previous filters, and the predicted mean $\hat{x}_{k}^\circ$ and covariance $P_{k}^\circ$ estimated from them. A similar process apply to the propagation of the uncertainty in observations, taking into account both the uncertainty in the system state and the sensor noise (refer to the two terms in the innovation covariance of DEKF above). The reader is referred to the original work [8] for the filter equations, not reproduced here for the sake of conciseness.

### 3.2 Dependent coordinates methods

Solving the position problem every time step has a high computational cost. For this reason, it is worth looking for alternatives to include the constraints into the Kalman filter.

#### 3.2.1 Discrete iterated extended Kalman filter with perfect measurements (DIEKFpm)

This method is a expansion of the standard DIEKF [7] to cope with constraints in its state space by employing so-called perfect measurements [4]. The key idea consists of augmenting the vector of observations $h(x)$ to include virtual observations that reflect the fulfillment of the kinematics constraints in both position and velocities. The augmented observation function $h'(x)$ is strongly nonlinear, hence the application in this case of an iterated estimator, capable of reducing the linearization errors to acceptable levels.

For the benchmark at hand, the state vector of this estimator comprises the multibody model coordinates and their derivatives, that is, $\hat{x}^\top = \{\hat{q}^\top, \hat{\dot{q}}^\top\}$. We define the augmented observation model $h'(\hat{x})$ as the concatenation of the real sensors $h(x)$ and the kinematic constraints in position and velocity, such as $h'(x)^\top = [h(x)^\top \Phi(x)^\top \Phi(x)^\top]$. This affects the calculation of the innovation (or "residual"), which must compare the actual sensor readings and current constraint errors with their predictions. For all time steps $k$ and iteration index $i$, the predicted values of the constrains are always zero, i.e.

$$y_{k,i} = \begin{bmatrix} o_k \\ 0_{2n \times 1} \end{bmatrix} - h'(\hat{x}_{k,i}) = \begin{bmatrix} o_k - h(\hat{x}_{k,i}) \\ -\Phi(\hat{q}_{k,i}) \\ -\Phi(\hat{\dot{q}}_{k,i}) \end{bmatrix}$$

The adjective "perfect" that names this method comes from the assumption that there is no error source in the virtual observations. In practice, this implies employing an extended sensor covariance matrix $\Sigma_k'$ with the structure:

$$\Sigma_k' = \begin{bmatrix} \Sigma_k & 0 \\ 0 & 0 \end{bmatrix}$$

#### 3.2.2 Extended Kalman filter with projections (projectionEKF)

Other method in dependent coordinates was also developed. The states of this method comprise the full set of position and velocity coordinates of the multibody model. The propagation and correction stages are performed as if the states were independent. After that, the states are projected over the constraint manifold, based in the constrained Kalman filter algorithms presented in [4]. However, some modifications were made to the algorithm to adapt it to the nature of the multibody simulations. The projections are performed in two stages. First, the position coordinates
are projected. As the position constraints are non-linear, the process is iterative, using the next
equation:
\[ \mathbf{q} = \mathbf{q}^* - \mathbf{W}^{-1} \Phi \Phi^T \mathbf{q}^* - \mathbf{W}^{-1} \Phi \Phi^T \mathbf{q}^* - 1 \Phi \Phi^T (\mathbf{q}^* - \mathbf{W}^{-1} \Phi \Phi^T \mathbf{q}^* - 1 \Phi \Phi^T \mathbf{q}^*) \quad (18) \]
where \( \mathbf{q}^* \) are the position coordinates after the correction stage, and \( \mathbf{W} \) is a weight matrix. After
that, the velocity coordinates are projected. As the constraint at velocity level are linear, this
projection is performed in one step:
\[ \dot{\mathbf{q}} = \dot{\mathbf{q}}^* - \mathbf{W}^{-1} \Phi \Phi^T \mathbf{q}^* - \mathbf{W}^{-1} \Phi \Phi^T \mathbf{q}^* - 1 \Phi \Phi^T (\mathbf{q}^* - \mathbf{W}^{-1} \Phi \Phi^T \mathbf{q}^* - 1 \Phi \Phi^T \mathbf{q}^*) \quad (19) \]
where \( \dot{\mathbf{q}} \) is the vector of velocities obtained after the correction stage of the Kalman filter.
After applying the projections, the covariance matrix of the states, \( \mathbf{P} \), must be updated to take into
account the effect of the projections, as follows:
\[ \mathbf{P_q} = \mathbf{P_q}^* - \mathbf{P_q}^* \Phi^T (\Phi \mathbf{P_q}^* \Phi^T)^{-1} \Phi \mathbf{P_q}^* \quad (20a) \]
\[ \mathbf{P_q} = \mathbf{P_q}^* - \mathbf{P_q}^* \Phi^T (\Phi \mathbf{P_q}^* \Phi^T)^{-1} \Phi \mathbf{P_q}^* \quad (20b) \]
where \( \mathbf{P_q}^* \) and \( \mathbf{P_q}^* \) are the blocks of of the covariance matrix correspondent to positions and
velocities before the projection process, and \( \mathbf{P_q}^* \) and \( \mathbf{P_q}^* \) are the blocks of the covariance matrix
after the projection process.
Nevertheless, it was found that this method does not work properly. The reason is that, in general,
the information from the sensors is related not to the full state of the mechanism, but only to
some coordinates. This fact produces that during the correction stage, the constraints are broken.
After that, during the projection process, the mechanism is conducted to a new state in which the
constrains are fulfilled, but most of the effect of the correction is lost. For this reason, this method
was not considered in the benchmark.

3.3 Mixed method: errorEKF
Most of the methods presented previously have some common drawbacks. One of them is that the
integrator must be explicit. Also, the nonlinearities of the multibody models can be not properly
handled by the EKF, which is only a first order approximation. Both these problems are overcome
by the UKF method, but at high computational cost. So a new formulation was developed. In it,
the multibody simulation is run in parallel with a Kalman filter which estimates not the position
nor the velocity vectors of the multibody model, but the errors of the independent coordinates and
velocities, so it is an error-state extended Kalman filter (also known as indirect extended Kalman
filter). With this approach, any existing multibody simulation can be used, regardless of the kind
of integrator or multibody formulation employed. After one step of the multibody simulation is
performed, the estimation of the error is launched. The propagation phase is performed following
the next equations:
\[ \dot{x}_k^- = 0 \quad (21) \]
\[ \mathbf{P}_k^- = \mathbf{f}_{x_k^-}^T \mathbf{P}_{x_k^-}^{-1} \mathbf{f}_{x_k^-} + \Sigma_{x_k^-} \quad (22) \]
These equations are the conventional equations for the propagation of the Kalman state. However,
as the estimated errors are fed back to the multibody simulation, the estimation of the error in the
propagation phase is always null, as shown in Eq. (21).
The equations for the correction phase of the filter are also equivalent to the ones found in a
conventional Kalman filter. After the correction stage, the estimation for the errors of positions
and velocities of the independent coordinates are obtained. However, to correct the state of the
multibody system, the errors for all the coordinates must be obtained, so them must be projected
over the constraints manifold.
An error in position means that an increment to the coordinates should be applied to get the position
corrected. Such an increment must fulfill the velocity constraints. So the increments applied
Figure 2. The final factor graph. Circle nodes are problem variables, rectangular nodes are factors.

to the coordinates of the mechanism are calculated by solving the velocity problem. This fact represent an advantage compared against the other methods based in independent coordinates in which solving the position problem is necessary, since the velocity problem is linear, and can be solved without employing iterative methods.

As said previously, the velocity problem is linear, so for correcting the velocities of the multibody system, the velocity problem must be solved using the estimation of the velocity error of the degrees of freedom. The result is added to the vector of velocities of the multibody model.

Although this method does not get to a perfect fulfilling of the constraints, it is enough to provide accurate estimations of the position and velocity of a mechanism.

3.4 The reference: a batch smoother

All state observers introduced up to this point share the key aspect of being filters. A filter processes the new information available at each timestep sequentially and immediately outputs an estimate of the current system state. Therefore, a clear advantage of filters is the fast availability of state estimates suitable for real-time closed-loop control, among other applications. On the other hand, the computational efficiency of filters is gained at the cost of forgetting the probability distribution of the entire history of the dynamic system except the very last timestep. If one uses Gaussian distributions to implement this process of forgetting information, named marginalization in the statistical inference literature [9], significant errors are introduced in the case of nonlinear dynamic systems, especially regarding the cross-covariances between the different variables.

The implication is that observation-based corrections of all the filters introduced above are sub-optimal in some degree due to inaccuracies in cross-correlation terms of the estimate covariance matrix. Additionally, some of those filters have another source of approximation in the first (EKF) or second-order (UKF) truncation of Taylor series for system equations.

As an alternative, we introduce in this work a novel smoother estimator which serves as a reference of the achievable accuracy for each combination of model and sensor errors in each benchmarked mechanism. Unlike filters, a smoother handles a joint probability distribution for the dynamic state during a sequence of timesteps. In particular, we have designed a batch estimator which considers the full history of dynamic states, which provides a better estimation than filters since the entire trajectory of the system is simultaneously subject to all existing constraints, both forwards and backwards in time, effectively smoothing the estimation as much as possible –hence the name. Our novel method is based on the Dynamic Bayesian Network (DBN) model of a Multibody Dynamic system devised in [5], and which is shown in Figure 2(a). A DBN encodes the causal relationships (directed edges) between all the variables (nodes) involved in a dynamic problem. Please refer to
for an in-depth discussion. In this work, we convert the DBN into its equivalent factor-graph form (see e.g. [10]), a kind of undirected bipartite graph where variables are represented as circular nodes, factor functions encode the relationships between variables (represented as square nodes) and undirected edges join factors with all the involved variables. Using the rules for converting DBNs into factor graphs we arrive at Figure 2(b). It is known that factor graphs allow writing down the full joint probability distribution of the system as the product of one cost function per factor node [9], such that in our case we arrive at:

$$\varphi(\hat{x}_0, \hat{z}_0, o_{0:t}, f_{0:t}, B) = \varphi_{\text{ini}}(\hat{x}_0)\varphi_B(B)\prod_{i=1}^{t} \varphi_o(\hat{z}_i, \hat{x}_i, \hat{x}_i, B)\varphi_{\text{com}}(\hat{z}_i, \hat{x}_i, f_i)\varphi_i(\hat{x}_i, \hat{x}_{i-1}, \hat{x}_i)$$  \hspace{1cm} (23)

Where each term has a well-defined physical meaning: $\varphi_{\text{ini}}$ represents a (typically loose) constraint regarding the a priori knowledge about the initial dynamical state of the system, $\varphi_B$ stands for the knowledge about the branch of the mechanism for those having multiple assembly configurations (please, refer to Figure 3 in [5]), $\varphi_o$ stands for the sensor models, $\varphi_{\text{com}}$ introduces the predicted accelerations according to the equations of motion and the theoretical multibody model, and $\varphi_i$ represents the constraints imposed by a numerical integrator to consecutive dynamic states.

Assuming Gaussian distributions for errors in all factors, it can be shown that the negative logarithm of Eq. 23 becomes a quadratic cost function which can be optimized numerically to obtain the most likely estimation of the mechanism dynamical trajectory. In this work we employed the Levenberg-Marquardt with excellent convergence results towards the optimal solution.

4 RESULTS

Four different configurations were tested in this work. For every one, two levels of modelization error were considered: 1 and $4 \, m/s^2$ of error in the gravity, and four different sampling rates of the sensors: one measurement every time step, and one measurement every 5, 10, and 15 time steps. The noise of the position sensors (encoders) employed in this work has a standard deviation of 1 deg, while the noise of the gyroscopes has a standard deviation of 1 deg/s.

4.1 Test 1: Five-bar linkage with gyroscopes

In this test, the mechanism to be considered is the the five-bar linkage. The sensors installed are two gyroscopes in both coupler links. In fig. 3 can be seen that the CEKF can only run in this configuration if it has a measurement every time step, and even in that case, the results are not very accurate. Next, filter in accuracy in the DIEKFpm. The three best filters are very close to each other, being the best the UKF, then the errorEKF, and in third place, the DEKF. The batch smoother provides the best results, as expected.

None of these methods could provide proper results with an error of $4 \, m/s$ in this configuration. Possibly, the reason for this fact is that, due to the particular geometry of the mechanism, in some ranges of the motion, big displacements of the crank correspond to small angular rates in the coupler rods, so the gyroscopes in those bars do not provide enough information to correct the position of the system.

4.2 Test 2: Five-bar linkage with encoders

This test is based in the same mechanism than that of the previous one, but with a different set of sensors. Now, one encoder is considered in each crank. The results can be seen in fig. 4.

With this configuration, bigger errors in the model can be handled. Also, the CEKF can be run with a sensor with a lower sampling rate than in the previous test. However, its performance is still the worst. The UKF, the errorEKF and the DEKF are even closer that in the previous test, with almost any difference in between them in terms of accuracy.
Figure 3. (a) Results for the five-bar linkage with two gyroscopes and $1 \, m/s^2$ of error in the gravity model. In the y axis, the RMS error in $rad/s$ is represented, while in the x axis is represented the sampling time of the sensors, expressed as number of time steps. (b) Legend

Figure 4. Results for the five-bar linkage with two encoders. (a) $1 \, m/s^2$ of error in the gravity model. (b) $4 \, m/s^2$ of error in the gravity model.

4.3 Test 3: Four-bar linkage with a gyroscope

The third test is based in the four-bar linkage, with a gyroscope installed on the coupler rod. The results can be seen in fig. 5

In this case, the geometry of the four-bar linkage allows the gyroscope to provide more information about the position of the mechanism than the five-bar linkage, so the observers remained stable even with the highest level of modeling error.

The CEKF presents problems again if the sampling rate of the sensors is not high enough. Regarding the other observers, it is remarkable that the DIEKFpm has the best accuracy if the measurement of the sensor is available at every time step, but when the sensor has a lower sampling rate, the results became worse, especially when the modeling error is high. Also, the batch smoother could not provide proper estimations in the test with the highest modeling error.

4.4 Test 4: Four-bar linkage with an encoder

This test is based in the four-bar linkage with an encoder in the left crank. The results are shown in fig. 6. In this test, the DIEKFpm obtained the best results for the lowest level of modeling error, but the worst after the CEKF with the highest modeling error. The CEKF was again the worst in both tests. Very similar accuracy was obtained from the UKF, the CEKF, and the errorEKF.

4.5 Performance

For an estate observer to be useful, it must be run in real time. Although the programs employed for this work were not designed nor implemented with performance in mind, it is expected that
the faster algorithm will perform faster both in the programs used here, or in a program designed and implemented specifically for real time performance. For this reason, a comparative of the time employed to run two of the previous tests is provided in fig. 7. The time varies depending on the type of test run, so this figure can be used just as a guide to see which method is faster.

It is worth focusing on the time of the DEKF, the UKF, and the errorEKF, because they got very similar results with regard the accuracy. However the computational cost of them is very different, being the errorEKF the fastest, and the UKF the slowest.

5 CONCLUSIONS

In this work, several state observers based on multibody models have been tested, checking the accuracy of the algorithms and their speed. Most of the methods tested were already published, while other, such as the errorEKF, the projectionEKF, and the batch smoother were developed in
this work. The batch smoother is not an observer. However, it uses the whole history of the states of the mechanism to obtain the most likely trajectory. It is used here as a reference for the best possible solution. The projectionEKF was not considered in the benchmark because it did not work as well as expected. However, the errorEKF showed an accuracy at the level of the best filters with a much lower computational cost.

The tests performed in this work were based in two planar mechanism: a four-bar and a five-bar linkages, with two different set of sensors each, and with four different sampling rates for the sensors.

Our reference MATLAB implementation has been released as Open Source\footnote{See https://github.com/MEEs/mkbds-matlab}.

6 ACKNOWLEDGMENTS

This work has been partially funded by fellowship BES-2013-063598 of Spanish Government, the project "DAVARBOT" (DPI 2011-22513) and the grant program JDC-MICINN 2011, financed by the Spanish Ministry of Science and Innovation.

REFERENCES


The Segway as a suitable multibody benchmark example

Olivier Verlinden∗, Michal Hajžman#, Pavel Polach†, Miroslav Byrtus#

∗Faculty of Engineering
Université de Mons – UMONS
Place du Parc, 20 – 7000 Mons, Belgium
olivier.verlinden@umons.ac.be

#Faculty of Applied Sciences
University of West Bohemia
Univerzitní 8, 306 14 Plzeň, Czech Republic
[mhajzman, mbyrtus]@kme.zcu.cz

†Section of Materials and Mechanical Engineering Research
Research and Testing Institute Plzeň
Tylova 1581/46, 301 00 Plzeň, Czech Republic
polach@vzuplzen.cz

ABSTRACT
The purpose of this paper is to propose a collection of benchmark multibody problems built around the so-called personal transporter, whose Segway is the most famous commercial implementation. The proposed benchmark deals with a rather simple system comprising only 5 bodies (the main body, 2 wheels and 2 motor rotors) and driven by 2 DC motors, but its main interest is its scalability. The simplest model is planar and considers that the wheels roll without any slip so that it involves only 2 DOF and its equations of motion can be derived manually. The motor action allows to envisage increasing complexity situations: no motor action (which makes the system conservative), feedback state regulation or feedback state position control. The feedback regulation only involves state dependent voltage while position control requires its own differential equation to integrate the tracking error. Besides, the mechanical complexity can be increased by modelling the wheel ground contact as a tire or by considering the 3D motion of the system.

The paper compares the simulation results related to the 2D system, obtained from (a) Matlab models developed according to minimal or Cartesian coordinates (b) the EasyDyn framework developed at the University of Mons, and the commercial codes (c) MSC/ADAMS and (d) alaska.

Keywords: benchmark, forward dynamics, equation of motion, control, software.

1 INTRODUCTION
The availability of benchmark problems is largely recognized to be necessary when dealing with simulation software. Hence, the IFToMM TC Multibody Dynamics decided to publish the Library of Computational Benchmark Problems [1] on its website. In this paper, we propose as a benchmark the so-called personal transporter, whose Segway is the most famous commercial implementation.

The model consists of only 5 bodies (Figure 1): the main body (denoted here by the passenger), two wheels, and rotors (including the gearbox) of the motors driving the wheels. However, the problem is interesting as several levels of complexity can be easily defined on the same basis, from a simple 2 DOF system (if the planar motion only is considered and the wheels are assumed to roll without any slip on the ground), to a 8 DOF system if wheels are modelled as tires in 3D. The equations of the simplest models will even be derived manually and integrated in a tool such as Matlab, Scilab or Octave. Moreover, the system is naturally unstable, like an inverted pendulum, and can only be simulated if a control is applied on the DC motors. Again, several levels will be considered: a simple state feedback regulator making the vehicle going back to a fixed reference position, or a state feedback controller imposing the longitudinal motion in the same time. It is of interest to note that the latter requires its own differential equation to integrate the position error. In the 3D case, the yaw motion is also to be controlled.

In the following, we will present simulation results obtained from various tools available in the laboratories of the authors. The considered packages, as well as the coordinates they are based
Table 1. The summary of possible types of Segway models and computational tools.

<table>
<thead>
<tr>
<th></th>
<th>2D model</th>
<th>2D model</th>
<th>3D model</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>perfect contact</td>
<td>tire</td>
<td>tire</td>
</tr>
<tr>
<td>EasyDyn [2] (C++, minimal coords.)</td>
<td>✓</td>
<td>✓</td>
<td>(✓)</td>
</tr>
<tr>
<td>MATLAB [7], minimal coords.</td>
<td>✓</td>
<td>✓</td>
<td>-</td>
</tr>
<tr>
<td>MATLAB, cartesian coords.</td>
<td>✓</td>
<td>(✓)</td>
<td>-</td>
</tr>
<tr>
<td>alaska [5] (relative coords.)</td>
<td>✓</td>
<td>(✓)</td>
<td>-</td>
</tr>
<tr>
<td>MSC.ADAMS [6] (cartesian coords.)</td>
<td>✓</td>
<td>(✓)</td>
<td>(✓)</td>
</tr>
</tbody>
</table>

2 DESCRIPTION OF THE SYSTEM

3 The system data

The values of the most important geometric parameters, illustrated on figure 1, are

\[ L = 0.9 \text{ m}, \quad r = 0.2 \text{ m}, \quad c = 0.4 \text{ m}, \quad a = 0.25 \text{ m}. \]

Table 2. Inertia properties

<table>
<thead>
<tr>
<th></th>
<th>Passenger (body 1)</th>
<th>Wheel (bodies 2 and 3)</th>
<th>Rotor (bodies 4 and 5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>mass (kg)</td>
<td>90</td>
<td>15</td>
<td>1</td>
</tr>
<tr>
<td>( I_{xx} \text{ (kg.m}^2)</td>
<td>21.6</td>
<td>0.2</td>
<td>1.2e-4</td>
</tr>
<tr>
<td>( I_{yy} \text{ (kg.m}^2)</td>
<td>21.6</td>
<td>0.3</td>
<td>1e-4</td>
</tr>
<tr>
<td>( I_{zz} \text{ (kg.m}^2)</td>
<td>2.5</td>
<td>0.2</td>
<td>1.2e-4</td>
</tr>
</tbody>
</table>

Table 3. Motor data

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Terminal resistance ( R )</td>
<td>0.35 ( \Omega )</td>
</tr>
<tr>
<td>Torque (speed) coefficient ( K_m ) (( K_b ))</td>
<td>0.085 Nm/A</td>
</tr>
<tr>
<td>Reduction ratio ( n )</td>
<td>25</td>
</tr>
<tr>
<td>Supply voltage (maximum available voltage)</td>
<td>72 V</td>
</tr>
</tbody>
</table>
Five different bodies can be considered: the main body (called the passenger), the two wheels, and the rotors of the DC motors (to whom the inertia of the rotating parts of the gearbox have been added) driving the wheels. Their inertia properties are summarized in table 2.

<table>
<thead>
<tr>
<th>Table 4. Tire data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outer radius r1</td>
</tr>
<tr>
<td>Torus radius r2</td>
</tr>
<tr>
<td>Radial stiffness Kz</td>
</tr>
<tr>
<td>Radial damping Cz</td>
</tr>
<tr>
<td>Nominal normal force Fznom</td>
</tr>
<tr>
<td>Long. stiffness Clongnom</td>
</tr>
<tr>
<td>Long. exponent nlong</td>
</tr>
<tr>
<td>Cornering stiffness C1atnom</td>
</tr>
<tr>
<td>Cornering exponent n1at</td>
</tr>
<tr>
<td>Camber stiffness Ccambernom</td>
</tr>
<tr>
<td>Camber exponent ncamber</td>
</tr>
<tr>
<td>Dynamic friction coeff. fClbd</td>
</tr>
</tbody>
</table>

Each wheel of the Segway is driven by a DC motor through a gearbox, the related data being gathered in table 3. Finally, the tire characteristics are given in table 4.

4 Simulation of the 2D 2DOF system

4.1 Equations of motion

If we consider only the motion in the XZ plane and assume that the wheels do not slip with respect to the ground, the system only has 2 degrees of freedom so that the kinematics can be completely defined in terms of the \( n_{cp} = 2 \) configuration parameters \( x \) and \( \theta \):

\[
q = \begin{bmatrix} x \\ \theta \end{bmatrix}
\]

The translational velocities of the bodies centers of mass (body 1=passenger, bodies 2 and 3=left and right wheels and bodies 4 and 5=left and right rotors) are given by

\[
\{v_1\} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \dot{x} + \begin{bmatrix} L \cos \theta \\ -L \sin \theta \end{bmatrix} \dot{\theta} \quad \{v_{2,3,4,5}\} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \dot{x} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} \dot{\theta}
\]

showing the partial translational velocities \( \mathbf{d}_{i,j} \) defined by \( \mathbf{v}_j = \sum_{j=1}^{n_{cp}} \mathbf{d}_{i,j} \cdot \dot{q}_j \).

In the same way, the rotational velocities of the bodies correspond to

\[
\{\omega_1\} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \dot{x} + \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \dot{\theta} \quad \{\omega_{2,3}\} = \begin{bmatrix} 0 \\ \frac{1}{r} \\ 0 \end{bmatrix} \dot{x} + \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \dot{\theta} \quad \{\omega_{4,5}\} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \dot{x} + \begin{bmatrix} 0 \\ 0 \\ 1-n \end{bmatrix} \dot{\theta}
\]

showing the partial rotational velocities \( \mathbf{\delta}_{i,j} \) defined by \( \omega_i = \sum_{j=1}^{n_{cp}} \mathbf{\delta}_{i,j} \cdot \dot{q}_j \).

The rotational velocity of the rotor is less straightforward. If \( n \) is the reduction ratio of the gearbox, it comes

\[
\omega_{rot/ground} - \omega_{pass/ground} = n \cdot (\omega_{wheel/ground} - \omega_{pass/ground})
\]

\[
\rightarrow \omega_{rot/ground} = n \cdot \omega_{wheel/ground} - (n-1) \cdot \omega_{pass/ground} = \frac{n}{r} \dot{x} - (n-1) \dot{\theta}
\]

The resultant force \( \mathbf{R}_i \) and moment \( \mathbf{M}_i \) of all applied forces exerted on each body \( i \) are expressed

\[
\{\mathbf{R}_1\} = \begin{bmatrix} 0 \\ 0 \\ -m_1 g \end{bmatrix} \quad \{\mathbf{R}_2\} = \{\mathbf{R}_3\} = \begin{bmatrix} 0 \\ 0 \\ -m_2 g \end{bmatrix} \quad \{\mathbf{R}_4\} = \{\mathbf{R}_5\} = \begin{bmatrix} 0 \\ 0 \\ -m_4 g \end{bmatrix}
\]

\[
\{\mathbf{M}_1\} = \begin{bmatrix} 0 \\ -2T_{mot} \\ 0 \end{bmatrix} \quad \{\mathbf{M}_2\} = \{\mathbf{M}_3\} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad \{\mathbf{M}_4\} = \{\mathbf{M}_5\} = \begin{bmatrix} 0 \\ T_{mot} \\ 0 \end{bmatrix}
\]
with $g$ the gravity acceleration and $T_{mot}$ the motor torque (for one motor). Joint forces (namely the ground contact force) are not taken into account as they do not develop any power.

Knowing the kinematics and the applied forces, the equations of motion can be constructed in the following way [2]

$$
M(q) \cdot \ddot{q} + h(q, \dot{q}) = g(q, \dot{q}, t)
$$

with

- $M$ the mass matrix of dimension $n_{cp} \times n_{cp}$, defined by
  $$
  M_{jk} = \sum_{i=1}^{n_{cp}} \left[ m_i \mathbf{d}^{i,j} \cdot \mathbf{d}^{i,k} + \Phi \iota_i \cdot (\Phi G_i \cdot \Phi \iota_i) \right]
  $$
  with $n_B$ the number of bodies, $m_i$ and $\Phi G_i$ the mass and the central inertia tensor of body $i$;
- $h$ a general term gathering the centrifugal and Coriolis terms
  $$
  h_j = \sum_{i=1}^{n_{cp}} \left[ m_i \mathbf{d}^{i,j} \cdot \left( \sum_{k=1}^{n_{cp}} \mathbf{d}^{i,k} \cdot \dot{q}_k \right) + \Phi \iota_i \cdot (\Phi \omega_i \cdot \Phi \omega_i) + \Phi \iota_i \cdot (\Phi G_i \cdot \Phi \omega_i) \right]
  $$
  and $g$ the contribution of the applied forces
  $$
  g_j = \sum_{i=1}^{n_{cp}} \left[ \mathbf{d}^{i,j} \cdot \Phi R_i + \Phi \iota_i \cdot \Phi M_i \right]
  $$

If we denote by $I_i$ the moment of inertia of body $i$ about the $Y$ axis, the development of the previous expressions leads to the 2 equations of motion of our model of Segway

$$
\left( m_1 + m_2 + m_3 + m_4 + m_5 + \frac{I_2 + I_3}{r^2} + n^2 \frac{I_4 + I_5}{r^2} \right) \ddot{x} + \\
\left( m_1 L \cos \theta + n (1-n) \frac{I_4 + I_5}{r} \right) \ddot{\theta} - m_1 L \sin \theta \ddot{\theta}^2 = 2T_{mot} \frac{n}{r}
$$

$$
\left( m_1 L \cos \theta + n (1-n) \frac{I_4 + I_5}{r} \right) \ddot{x} \left( m_1 L^2 + I_1 + (I_4 + I_5)(1-n)^2 \right) \ddot{\theta} = m_1 g L \sin \theta - 2n T_{mot}
$$

4.2 Matlab model

The equations of motion have been implemented under Matlab. For that purpose, they were transformed to the equivalent first-order Cauchy form

$$
\mathbf{v} = M^{-1} \left( g(q, \mathbf{v}, t) - h(q, \mathbf{v}) \right) \quad \dot{q} = \mathbf{v}
$$

The matrices $M$, $g$ and $h$ are constructed according to the formulation presented before, by taking advantage of the matrix algebra of Matlab. For the simulation, the ode45 method has been selected. The relative and absolute tolerances were set to $10^{-6}$.

4.3 EasyDyn model

EasyDyn is a general framework for the simulation of dynamical systems and, more particularly, multibody systems. The interested reader will find more details in [2]. Actually, EasyDyn provides on one hand a C++ library which constructs and integrates the equations of motion from the kinematics and the external forces, and a symbolic tool called CAGeM, generating a basic C++ EasyDyn application from inertia data and position expressions related to each body.

For the sake of illustration, the kinematics of the system is described in the following way

\[
\begin{align*}
T_{0G}[1] &= T_{disp}(q[0], 0, x) \cdot T_{roty}(q[1]) \cdot T_{disp}(0, 0, L); \\
T_{0G}[2] &= T_{disp}(q[0], c, x) \cdot T_{roty}(q[0]) / x; \\
T_{0G}[3] &= T_{disp}(q[0], -c, x) \cdot T_{roty}(q[0]) / x; \\
T_{0G}[4] &= T_{disp}(q[0], a, x) \cdot T_{roty}(n \cdot (q[0] / x - q[1]) + q[1]); \\
T_{0G}[5] &= T_{disp}(q[0], -a, x) \cdot T_{roty}(n \cdot (q[0] / x - q[1]) + q[1]);
\end{align*}
\]

where $q[0]$ and $q[1]$ correspond to $x$ and $\theta$ respectively.

The integration method in EasyDyn is Newmark. The time step is automatically adjusted to keep the error below $10^{-6}$. 

1610
4.4 Matlab model based on Cartesian coordinates

A multibody simulation tool under Matlab was created at The University of West Bohemia in Plzen. It is characterized by the usage of Cartesian (physical) coordinates [3], which leads to a system of differential-algebraic equations. The so-called underlying equation of motion is expressed by eliminating Lagrange multipliers and it is integrated using standard numerical solvers for ODEs in Matlab. Baumgarte’s stabilization is employed [4] in order to improve the numerical solution. The whole Segway model is composed of five bodies (passenger, two wheels, two motors). Two wheels can be considered to be one body with respect to the symmetry of the problem and the motion as well as two motors. Four kinematic constraints were defined in the model — one constraint for idealized rolling of wheels, two revolute constraints between the wheel and the passenger and between the motor and the passenger and one constraint is coupling the rotational motion between all three bodies. For the simulation, the ode45 method has been selected with the relative and absolute tolerances set to $10^{-6}$.

4.5 Alaska model

Alaska (advanced lagrangian solver in kinetic analysis) [5] is a simulation tool for the analysis, the synthesis and the optimization of mechatronical systems. The implemented multibody formalism is based on relative coordinates, thus it is necessary to pay attention to the proper kinematic tree of a studied system. The Segway model is created using five rigid bodies considering the symmetry of the system as mentioned earlier. Further one additional dummy body is used in order to separate the general rolling motion of the wheels into translational and rotational motion with one rolling constraint. Another kinematic constraint is relating the rotation of wheels, motors and passenger. Numerical integration by Shampine-Gordon method [5] was chosen with tolerances set to $10^{-6}$.

4.6 ADAMS model

The structure of the Segway model built in the well-known MSC.ADAMS software [6] is the same as described for Matlab model based on Cartesian coordinates. The rolling and driving kinematic constraints were implemented using GCON function. For the simulation, the GSTIFF method has been selected with the relative and absolute tolerances set to $10^{-6}$.

4.7 Simulation of the free system

4.8 Simulation results

The free motion of the personal transporter has been simulated from the following initial conditions

$$ x_0 = 0 \text{ m} \quad \theta_0 = 0.2 \text{ rad} \quad \dot{x}_0 = 0 \text{ m/s} \quad \dot{\theta}_0 = 0 \text{ rad/s} $$

The time history of the configuration parameters is shown in figure 2. The curves perfectly match. Figure 3 presents the total energy of the system, which should be constant. The MSC/ADAMS and EasyDyn models exhibit variations up to 0.2% and 0.02% respectively. Let us recall that the integration tolerances have been tightened down to $1e^{-6}$.

5 REGULATED MODEL

5.1 Actuator modelling

We assume here that the personal transporter is driven by permanent magnet DC motors. If we assume that the inductance effect is negligible, the electrical equation of the motor reads

$$ u = R \cdot i + K_b \omega_{rot/\text{pass}} \quad (12) $$

with $u$ the supply voltage, $R$ the terminal resistance, $i$ the current, $K_b$ the electromotive feedback constant, from which we can get the motor torque.
In the following, the supply voltage $u$ is the input (or command) variable used to stabilize/control the system.

5.2 Steps to build the controllers

In a first step, the equations of motion are linearized about a configuration corresponding to the vehicle at rest and vertical ($\theta=0$). The resulting equations are written

$$M \Delta \ddot{q} + C \Delta \dot{q} + K \Delta q = Fu$$  \hspace{1cm} (14)$$

where $M$, $C$ and $K$ are the mass, damping and stiffness matrices respectively, $F$ the influence matrix and $q$ the input, which corresponds here to the supply voltage applied to each motor.

In the second step, the matrices are used to build the equivalent state-space form of the system equations

$$\dot{x} = Ax + Bu$$  \hspace{1cm} (15)$$

according to

$$\dot{x} = \begin{bmatrix} \dot{q} \\ \dot{q} \end{bmatrix} \quad A = \begin{bmatrix} 0 & I \\ -M^{-1}K & -M^{-1}C \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ M^{-1}F \end{bmatrix}$$  \hspace{1cm} (16)$$

where $0$ and $I$ are zero and identity matrices of appropriate dimensions.
The number of states, denoted by $n$, is equal to twice the number of configuration parameters, that’s to say 4.

In the 3rd step, a linear state-feedback strategy is applied, consisting in applying a voltage of the form

$$u = -K \cdot x = -K_x \dot{x} - K_\theta \dot{\theta}$$

so as to stabilize the system.

As a first option, the state feedback gain vector $K$ can be constructed by pole placement, i.e. so that the evolution matrix of the closed system $A - BK$ has the eigenvalues selected by the designer. It is easily computed from the place command of the Matlab control toolbox.

The second option consists in tuning the state feedback gain vector so as to minimize the following quadratic cost function

$$\tilde{J} = \int_0^\infty (x^T Q x + u^T Q_u u) dt$$

where $Q_x \geq 0$ and $Q_u > 0$ are symmetric, positive semi-definite matrices of the appropriate dimensions. The corresponding optimal feedback gain vector can be computed by the single command lqr under Matlab.

Let us mention that the controller design is made from the Matlab minimal coordinates model and is then available to the benchmark participant.

5.3 Simulation results

A state feedback gain vector has been computed by pole placement so that the eigenvalues are worth

$$p_{1,2} = 1.1 \left( -\xi \omega_0 \pm j\sqrt{1 - \xi^2 \omega_0} \right) \quad p_{3,4} = 1.2 \left( -\xi \omega_0 \pm j\sqrt{1 - \xi^2 \omega_0} \right)$$

with $\omega_0 = 2\pi$ (1 Hz) and $\xi = 0.9$. The corresponding vector is

$$K_x = -350.95 \quad K_\theta = -640.72 \quad K_i = -185.81 \quad K_\dot{\theta} = -205.65$$

For the linear quadratic minimization, with a matrix $Q_x$ equal to the identity matrix and the $Q_u$ matrix set to 0.001, the resulting state feedback gain vector is equal to

$$K_x = -31.623 \quad K_\theta = -304.28 \quad K_i = -70.716 \quad K_\dot{\theta} = -113.44$$

The simulation results for the LQG regulator are presented in figure 4. A little disagreement can be observed on the maximum displacement, which is not fully explained. A reason could be the number of digits retained in the gain vector which is known to be very sensitive.
6 CONTROLLED SYSTEM

In the previous section, the vehicle was just stabilized but there is still no control on the position. For that purpose, a state-space feedback controller is designed according to the scheme presented in figure 5. The system is augmented so as to introduce a new state variable which is the integral of the error between the output variable \( y \) (\( x \) here) and the target \( r \). The presence of the integral action on the error allows to capture the properties of PID controllers [8]. The state-space form of the corresponding augmented system is written

\[
\dot{\mathbf{x}}^* = A^* \mathbf{x}^* + B^* u + E^* r \quad y = C^* \mathbf{x}^*
\]

in which \( \mathbf{x}^* \) is the augmented state vector and \( A^*, B^*, C^*, E^* \) the matrices of the augmented system, built from the original one as

\[
\mathbf{x}^* = \begin{bmatrix} \mathbf{x} \\ e \end{bmatrix}, \quad A^* = \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix}, \quad B^* = \begin{bmatrix} B \\ 0 \end{bmatrix}, \quad E^* = \begin{bmatrix} 0 & 1 \end{bmatrix}, \quad C^* = \begin{bmatrix} C & 0 \end{bmatrix}^T
\]

**Figure 5.** Closed loop state feedback with integral action

In our case the output variable \( y \) is the position \( x \) of the segway. If we denote by \( r(t) \) the wanted time history of \( x \), the previous model is completed by the following equation

\[
\dot{e} = x - r(t)
\]

The same methods can be used to build the state feedback control being now expressed as

\[
u = -K \cdot \mathbf{x} = -K_x x - K_{\theta} \theta - K_{\dot{x}} \dot{x} - K_{\dot{\theta}} \dot{\theta} - K_e e
\]

6.1 Simulation results

A state feedback gain vector has been computed by pole placement so that the eigenvalues are worth

\[
p_{1,2} = 0.5 \left(-\xi \omega_0 \pm j \sqrt{1 - \xi^2} \omega_0\right) \quad p_{3,4} = 0.6 \left(-\xi \omega_0 \pm j \sqrt{1 - \xi^2} \omega_0\right) \quad p_5 = 0.7 \ast \omega_0
\]

with \( \omega_0 = 2\pi \) (1 Hz) and \( \xi = 0.9 \). The corresponding vector is

\[
K_x = -111.0109 \quad K_\theta = -251.1743 \quad K_{\dot{x}} = -82.1588 \quad K_{\dot{\theta}} = -88.2848 \quad K_e = +86.8775
\]

For the linear quadratic minimization, the \( Q_x \) and \( Q_u \) matrices have been set to the identity matrix and 0.001 respectively, leading to the following state feedback gain vector

\[
K_x = -82.2695 \quad K_{\dot{x}} = -394.1588 \quad K_{\dot{x}} = -101.8294 \quad K_{\dot{\theta}} = -144.3819 \quad K_e = 31.6228
\]

The system has been simulated from a an initial position at rest, with the segway in vertical configuration. After 1 second, the target position is set to 1 m and the segway moves. This simulation can be considered as a step response. The simulation results for the LQG controllers are presented in figure 6. Some discrepancy can be observed on the maximum values of the angle, especially for Adams.
7 Simulation of the 2D system with tires

7.1 Equations of motion

The assumption that the wheels do not slip with respect to the ground can be released by defining the wheel/ground contact by a tire force element. In this case, the vertical and rotational motions of the wheel must be added and the kinematics of the system can be described by $n_{cp} = 4$ configuration parameters

$$q = \begin{bmatrix} x \\ z \\ \theta \\ \theta_{wr} \end{bmatrix}$$

with $z$ the Z coordinate of the center of the wheel and $\theta_{wr}$ the relative angle of the wheel (about Y) with respect to the passenger.

The translational and rotational velocities of the bodies become

$$\{v_1\} = \begin{bmatrix} 1 & 0 & L \cos \theta & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & -L \sin \theta & 0 \end{bmatrix} \cdot \begin{bmatrix} \dot{x} \\ \dot{z} \\ \dot{\theta} \\ \dot{\theta}_{wr} \end{bmatrix}$$

$$\{\omega_1\} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \ddot{x} \\ \ddot{z} \\ \ddot{\theta} \\ \ddot{\theta}_{wr} \end{bmatrix}$$

(29)

$$\{v_2\} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \dot{x} \\ \dot{z} \\ \dot{\theta} \\ \dot{\theta}_{wr} \end{bmatrix}$$

$$\{\omega_2\} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \ddot{x} \\ \ddot{z} \\ \ddot{\theta} \\ \ddot{\theta}_{wr} \end{bmatrix}$$

(30)

$$\{v_3\} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \dot{x} \\ \dot{z} \\ \dot{\theta} \\ \dot{\theta}_{wr} \end{bmatrix}$$

$$\{\omega_3\} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & n \\ 0 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \ddot{x} \\ \ddot{z} \\ \ddot{\theta} \\ \ddot{\theta}_{wr} \end{bmatrix}$$

(31)

Concerning the external forces, the longitudinal $F_{cx}$ and vertical $F_{cz}$ tire contact forces must be taken into account in the wrench applied to body 2, which now reads

$$\{R_2\} = \{R_3\} = \begin{bmatrix} F_{cx} \\ 0 \\ -m_2 g + F_{cz} \end{bmatrix}$$

$$\{M_2\} = \{M_3\} = \begin{bmatrix} 0 \\ -z F_{cx} \\ 0 \end{bmatrix}$$

(32)
A simplified tire model, corresponding to the model of the University of Arizona [9, 10] when only longitudinal slips are considered will be used. The forces are computed according to the following steps:

- The vertical force is computed by considering that the tire behaves radially like a spring-damper system of stiffness $K_z$ and damping coefficient $C_z$
  \[
  \text{if } z < r \quad \rightarrow \quad F_{cz} = K_z(r-z) - C_z\dot{z}
  \] (33)

- The longitudinal slip is determined by
  \[
  v_x = \frac{\dot{x} - (\dot{\theta} + \dot{\theta}_{av})r_e}{\max(|\dot{x}|,|\dot{\theta} + \dot{\theta}_{av}|r_e, 1e^{-3})}
  \] (34)
  with $r_e$ the effective rolling radius given by
  \[
  r_e = \frac{2}{3}z + \frac{1}{3}r
  \] (35)
  and $1e^{-3}$ a threshold to avoid a division by zero.

- The limit slip $v_{lim}$ and the effective friction coefficient $f$ are calculated according to
  \[
  v_{lim} = \frac{3f_df_c}{C_l} \quad f = f_s - (f_s - f_d) \frac{v_x}{v_{lim}}
  \] (36)
  where $C_l$ is the longitudinal stiffness of the tire, and $f_d$ and $f_s$ the dynamic and static friction coefficients respectively.

- And the longitudinal force is finally computed by
  \[
  F_{cx} = -C_l v_x \left(1 - \frac{C_l}{3f_cF_z} v_x \left(1 - \frac{C_l}{9f_cF_z} v_x \right)\right)
  \] (37)

The equations of motion can be constructed numerically under Matlab by following the same approach as for the 2 DOF model.

7.2 Developed Models

Basically, the previous models can be extended to include the tires:

- The minimal coordinates Matlab model just comprises 2 supplementary degrees of freedom, and the computation of the tire contact forces as developed in the previous section. The simulation involves 8 (9 for the controlled system) first order equations. As the equations are stiff, the ode45 integration method becomes completely inefficient and is replaced by ode15s.

- The EasyDyn model is actually the 3D model, the motion remaining 2D as the same voltage is applied on both motors.

- Two contact constraints in Matlab model based on Cartesian coordinates are replaced by forces representing tires.

- Similarly as in EasyDyn, the MSC/Adams model can be generally considered as the 3D model with the tires modelled using tire force elements.

The controllers are kept the same as the ones defined with the 2D system. The behaviour of the tire can be considered as some kind of disturbance and permits to assess its robustness.
7.3 Simulation results

Figure 7 shows the simulation results obtained for the free motion, superimposed with the response of the system with a perfect contact. The beginning of the fall is similar with and without tires but later, a major difference appears due to the tire deflection and the friction effects. Otherwise, models with tires produce consistent results although they no longer match as well as previously.

Figure 8 shows the time history of the X coordinate for the simulation of the regulated and controlled systems, when the controller is built by LQG. In this case, all models match quite well, due to the fact that the controller is rather soft, so that the energy losses in the tire are limited. Let us mention that the regulator/controller obtained by pole placement cannot stabilize the system as the necessary contact forces are over the friction limit.

8 3D model with tires

No results will be presented as the reference simulation still has to be defined for this model. The system still consists of 5 bodies but owns 8 DOF, 6 for the passenger which is free and one for each wheel. Some other modifications will be necessary for the controller as the yaw must also be controlled.

Let us note that the 3D model with wheels rolling without slip can also be of interest as this system is not holonomic: it owns 3 degrees of freedom but needs 5 configuration parameters which are linked by 2 constraints at velocity level. Consequently, it cannot be addressed with
9 CONCLUSIONS

As an answer to the need of fully defined benchmark problems in the multibody community, we have addressed in this paper the simulation of a personal transporter. We are quite confident in the presented results as the simulations were driven by the 4 authors, each of them on a different tool (M. Byrtus: Adams, M. Hajzman: Cartesian coordinates under Matlab, P. Polach: alaska and O. Verlinden: EasyDyn and Minimal coordinates under Matlab).

The presented system offers some interesting characteristics

- The benchmark participant can validate his model progressively, starting from a very simple 2 DOF mechanical system, for which a Matlab model is available, and evolving to more sophisticated models on the mechanical or control points of view.
- Several other features are possibly included in the test: conservation of energy (i.e. integration quality) for the simplest system, tire model including deformation and friction limit, supplementary differential equations related to the control.

The whole set of results cannot be presented here due to lack of place but will be made available on the benchmark web site.

REFERENCES


Validation of Flexible Multibody Dynamics
Beam Formulations using Benchmark Problems

Olivier A. Bauchau†, Peter Betsch‡, Alberto Cardona§, Johannes Gerstmayr¶, Ben Jonker‖, Pierangelo Masarati¶, Valentin Sonneville‖

†Hong Kong Univ. of Sci. and Tech., Clear Water Bay, Hong Kong
olivier.bauchau@ust.hk
‡Karlsruhe Institute of Technology, Karlsruhe, Germany
Peter.Betsch@kit.edu
§CIMEC (UNL / Conicet), Santa Fe, Argentina
cardona@intec.unl.edu.ar
¶Leopold-Franzens Universität Innsbruck, Innsbruck, Austria
Johannes.Gerstmayr@uibk.ac.at
‖University of Twente, Enschede, The Netherlands
j.b.jonker@utwente.nl
¶Politecnico di Milano, Milano, Italy
pierangelo.masarati@polimi.it
‡Université de Liège, Liège, Belgium
v.sonneville@ulg.ac.be

1 ABSTRACT

As the need to model flexibility arose in multibody dynamics, the floating frame of reference formulation was developed but this approach can yield inaccurate results when elastic displacements becomes large. While the use of three-dimensional finite element formulations overcomes this problem, the associated computational cost is overwhelming. Consequently, beam models, which are one-dimensional approximations of three-dimensional elasticity, have become the workhorse of many flexible multibody dynamics codes. Numerous beam formulations have been proposed, such as the geometrically exact beam formulation or the absolute nodal coordinate formulation, to name just two. New solution strategies have been investigated as well, including the intrinsic beam formulation or the DAE approach. This paper provides a systematic comparison of these various approaches, which will be assessed by comparing their predictions for four benchmark problems. The first problem is the Princeton beam experiment, a study of the static large displacement and rotation behavior of a simple cantilevered beam under a gravity tip load. The second problem, the four-bar mechanism, focuses on a flexible mechanism involving beams and revolute joints. The third problem investigates the behavior of a beam bent in its plane of greatest flexural rigidity, resulting in lateral buckling when a critical value of the transverse load is reached. The last problem investigates the dynamic stability of a rotating shaft. The predictions of eight independent codes are compared for these four benchmark problems and are found to be in close agreement with each other and with experimental measurements, when available.

Keywords: Benchmark problems, Flexible multibody systems

2 Introduction

Multibody dynamics was originally developed to deal with simple tree-like topologies composed of rigid bodies. As the need to model flexibility arose, the floating frame of reference formulation [1] was proposed. Unfortunately, as the magnitude of the elastic displacements increases, this formulation becomes increasingly inaccurate and the multibody dynamics community began to turn its attention to finite element based formulations. Simo [2] proposed the geometrically exact beam formulation (GEBF), which corrected the shortcomings of earlier co-rotational formulations [3]. In recent years, the absolute nodal coordinate formulation (ANCF) developed by Shabana et al. [4, 5] has received considerable attention for the modeling of flexible multibody systems. The GEBF and ANCF are easier to derive than the floating frame of reference approach and involve fewer assumptions, but little effort has been devoted to the systematic comparison of these two approaches. Romero [6] has presented a comparison of both qualitative and quantitative aspects of the two approaches. He concluded that the ANCF is more straightforward, while GEBF involves thorny
issues, such as the treatment of finite rotation. Unfortunately, the ANCF suffers from a number of locking mechanisms that must be eliminated to obtain accurate results. As pointed out by Gerstmayr, this can be accomplished in a number of ways, but the proposed techniques complicate the description of elastic forces, leading to more arduous implementations and moving away from the simplicity of the initial implementation. In some of the examples treated by Romero, the ANCF and GEBF did not converge to the same solution. In all cases, the computational efficiency of the GEBF was found to be superior to that of the ANCF.

Bauachau et al. [7] further compared the GEBF and ANCF to identify the causes of their differing computational efficiencies. First, they performed a kinematic analysis, in which the exact nodal displacements were prescribed and the predicted displacement and strain fields were compared for the two methods. The accuracies of the predicted strain fields were found to differ markedly: the predictions of the GEBF were more accurate than those of the ANCF. They attributed this phenomenon to the fact that the curvature field is obtained as a second derivative of the displacement in the ANCF, but as a first derivative only for the GEBF. Next, they carried out a static analysis to determine the solution of the problem. For the GEBF, the predictions of the static analysis are far more accurate than those obtained from the kinematic solution; in contrast, the same order of accuracy was obtained for the two solution procedures when using ANCF. In all cases, they reported that the predictions of the GEBF were more accurate than those of the ANCF. A further study by Bauachau et al. [8] compared the predictions of the two methodologies against experimental measurements.

Besides these various formulations of beam problems, novel solution strategies have also been investigated. For instance, Hegemier [9] and Hodges [10] have developed intrinsic formulations for beams. By eliminating the displacement and rotation fields from the geometrically exact equations of the problem, quadratic equations result and this feature simplifies the solution process. Cardona and Géradin [11] adapted the well known Hilber-Hughes-Taylor [12] time integration scheme initially developed for the finite element method to the finite rotation problems found in geometrically exact beams. Betsch and co-workers [13] proposed the use of the direction cosine matrix to represent the rotation tensor; this lead to differential-algebraic equations (DAE), which they solved using novel integration techniques. Finally, it is worth mentioning the increased application of Lie group techniques in rigid and flexible multibody dynamics [14]. The resulting solution techniques aim at preserving the invariant manifolds that characterize multibody systems.

3 The four benchmark problems

This study focuses on four benchmark problems: the Princeton beam experiment, the four-bar mechanism, the lateral buckling of a beam, and the stability of a rotating shaft described in sections 3.1, 3.2, and 3.3, respectively. For each benchmark problem, beam sectional properties such as bending, shearing, and torsional stiffnesses must be evaluated. In this effort, these properties were computed from the section’s geometric and stiffness properties using the procedure developed by Bauachau and Han [15, 16].

3.1 The Princeton beam experiment

The Princeton beam experiment [17, 18] is a study of the large displacement and rotation behavior of a simple cantilevered beam under a gravity tip load. A straight aluminum (T 7075) beam of length \( L = 0.508 \text{ m} \) with a rectangular cross-section of thickness \( t = 3.175 \text{ mm} \) and height \( h = 12.7 \text{ mm} \) was cantilevered at its root and subjected to a static concentrated transverse load \( P \) at its tip.

Figure 1 shows an end-view of the test set-up. An inertial frame of reference is selected as \( \mathcal{I} = ( \mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3 ) \) and material frame \( \mathcal{B} = ( \mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3 ) \) is attached at the beam’s root section, which is cantilevered into a bearing that allows rotation of the beam about its reference axis by an angle \( \theta \), called the “loading angle.” The gravity load applied at the beam tip is acting in the opposite direction of unit vector \( \mathcal{I}_3 \). Variation of the loading angle from 0 to 90 degrees yields
a range of nonlinear problems where torsion and bending in two directions are coupled.

Experimental results [17] consist of measurements of the beam’s tip deflection along the material unit vectors \( \mathbf{b}_2 \) and \( \mathbf{b}_3 \), denoted \( u_2 \) and \( u_3 \), respectively, and called the “flapwise” and “chordwise displacements,” respectively. The beam’s tip twist was also measured. Let \( \mathbf{R}^E = \{ \mathbf{b}_2^E, \mathbf{b}_5^E, \mathbf{b}_3^E \} \) denote the rotation tensor characterizing the orientation of the beam’s tip cross-section. In the absence of tip load, \( \mathbf{R}^E(P = 0) = \{ \mathbf{b}_2^E, \mathbf{b}_5^E, \mathbf{b}_3^E \} \), where \( \mathbf{b}_5^E = \{ 0, \sin \theta, \cos \theta \} \), and it then follows that \( \theta = \arctan(\mathbf{R}^E(P = 0)/\mathbf{R}^E(P = 0)) \). Under tip load \( P \), the orientation of the tip section is defined as \( \arctan(\mathbf{R}^E_2(P)/\mathbf{R}^E_3(P)) \) and the beam’s tip twist is defined as

\[
\phi = \arctan(\frac{\mathbf{R}^E_2}{\mathbf{R}^E_3}) - \theta. \tag{1}
\]

The procedure used to measure the twist angle experimentally is detailed in the report by Dowell and Traybar [17]. Data was acquired at loading angles of \( \theta = 0, \pm 15, \pm 30, \pm 45, \pm 60, \pm 75, \pm 90 \), and 180 degrees. For a perfect system, symmetry implies that the absolute values of the tip displacements and twist should be identical for loading angles \( \pm \theta \). In the experimental setting, these measurements differed, providing an estimate of their accuracy. Three loading conditions were used, \( P_1 = 4.448 \text{ N} \), \( P_2 = 8.896 \text{ N} \), and \( P_3 = 13.345 \text{ N} \).

The linear solution of the problem is found using the shear deformable beam theory described in structural analysis textbooks [19]. The tip transverse displacements are

\[
u^T = \left[ \frac{PL^3}{3H_3} + \frac{PL}{K_{22}} \right] \sin \theta, \quad u^T = \left[ \frac{PL^3}{3H_2} + \frac{PL}{K_{33}} \right] \cos \theta, \tag{2}
\]

where \( H_{22} \) and \( H_{33} \) are the bending stiffnesses about material unit vectors \( \mathbf{b}_2 \) and \( \mathbf{b}_3 \), respectively, and \( K_{22} \) and \( K_{33} \) the shearing stiffnesses along the same unit vectors, respectively. Of course, for linear theory, the tip twist vanishes.

For \( \theta = 0 \) or 180 and \( \theta = \pm 90 \), the beam undergoes planar deformation and elementary formulae of Timoshenko beam theory (2) provide the tip deflection in the linear regime. Using the Young’s modulus of T 7075 aluminium as \( E = 71.7 \text{ GPa} \) and Poisson’s ratio \( \nu = 0.31 \), hand calculations yield \( u^T = 5.004 \) and \( u^T = 80.34 \text{ mm} \) for the chordwise and flapwise tip displacements, respectively, at loading level \( P \). This compares favorably with experimental measurements of \( u^T = 5.3594 \) and \( u^T = 77.635 \text{ mm} \), respectively, resulting in -6.6% and +3.1% error, respectively. In this effort, the dimensions of the cross-section were adjusted slightly to achieve good correlation between measurements and predictions of linear theory in these two cases. The following data was used: \( L = 0.508 \text{ m}, t = 3.2024 \text{ mm}, h = 12.377 \text{ mm}, E = 71.7 \text{ GPa}, \nu = 0.31, \) and \( G = E/2(1 + \nu) = 27.37 \text{ GPa} \). These physical properties translate to the sectional stiffness properties listed in table 1 and the sectional mass properties per unit span is \( m_{00} = 0.1062 \text{ kg/m} \).

<table>
<thead>
<tr>
<th>Beam</th>
<th>Axial Shearing</th>
<th>Shearing Torsional</th>
<th>Bending</th>
<th>Bending</th>
</tr>
</thead>
<tbody>
<tr>
<td>S [MN]</td>
<td>K_{22} [MN]</td>
<td>K_{33} [MN]</td>
<td>H_{11} [N-m²]</td>
<td>H_{22} [N-m²]</td>
</tr>
<tr>
<td>2.842</td>
<td>0.6401</td>
<td>0.9039</td>
<td>3.103</td>
<td>36.28</td>
</tr>
</tbody>
</table>

**3.2 The four-bar mechanism**

Figure 2 depicts a flexible four-bar mechanism. Bar 1 is of length 0.12 m and is connected to the ground at point A by means of a revolute joint. Bar 2 is of length 0.24 m and is connected to bar 1
at point B with a revolute joint. Finally, bar 3 is of length 0.12 m and is connected to bar 2 and the ground at points C and D, respectively, by means of two revolute joints.

In the reference configuration, the bars of this planar mechanism intersect each other at 90 degree angles and the axes of rotation of the revolute joints at points A, B, and D are normal to the plane of the mechanism. To simulate an initial defect of the mechanism, the axis of rotation of the revolute joint at point C is rotated by +5 degrees about unit vector $\vec{i}_2$ indicated in fig. 2. The angular velocity of bar 1 at point A is prescribed as $\Omega = 0.6 \text{ rad/s}$ for the duration of the simulation.

Bars 1 and 2 are of square cross-section of size 16 by 16 mm; bar 3 has a square cross-section of size 8 by 8 mm. The three bars are made of steel, whose mechanical characteristics are Young’s modulus $E = 207 \text{ GPa}$ and Poisson’s ratio $\nu = 0.3$. These physical properties translate to the sectional stiffness properties listed in table 2. The sectional mass properties are as follows: mass per unit span $m_{00} = 1.997$ and 0.4992 kg/m, moments of inertia per unit span $m_{22} = m_{33} = 42.60$ and 2.662 mg·m²/m for bars 1 and 2, and bar 3, respectively.

If the bars were infinitely rigid, no motion would be possible because the mechanism locks. For elastic bars, motion becomes possible, but generates large, rapidly varying internal forces and moments. If the axes of rotation of the four revolute joints were orthogonal to the plane of the mechanism, the response of the system would be purely planar, and bars 1 and 3 would rotate at constant angular velocities about points A and D, respectively. The initial defect in the mechanism causes a markedly different response. Bar 1 rotates at the constant prescribed angular velocity, but bar 3 now oscillates back and forth, never completing an entire revolution.

| Table 2. Sectional stiffness properties of the bars |
|-----------------|-------|-------|-------|-------|-------|-------|
|                 | Axial | Shearing | Shearing | Torsional | Bending | Bending |
|                 | $S$ [MN] | $K_{22}$ [MN] | $K_{33}$ [MN] | $H_{11}$ [N·m²] | $H_{22}$ [N·m²] | $H_{33}$ [N·m²] |
| Bar 1 & 2       | 52.99 | 16.88  | 16.88  | 733.5  | 1131   | 1131   |
| Bar 3           | 13.25 | 4.220  | 4.220  | 45.84  | 70.66  | 70.66  |

3.3 Lateral buckling of a thin beam

If a beam is bent in its plane of greatest flexural rigidity, lateral buckling will occur when a critical value of the transverse load is reached. In this benchmark problem, the tip of a beam is subjected to a transverse load applied through a crank and link mechanism, as depicted in fig. 3. The beam is clamped at one end, while the other end is connected to the link through a spherical joint. The crank and link are modeled by flexible beams connected by revolute joints. As the crank rotates, the beam tip is pushed up. When the buckling load is reached, the beam snaps laterally and becomes significantly softer in bending due to the pronounced twisting deformation.

Figure 3 depicts the configuration of the problem. The beam is of length $L = 1 \text{ m}$, the crank and link lengths are $L_c = 0.05 \text{ m}$ and $L_d = 0.25 \text{ m}$, respectively. The rotation of the crank is prescribed as $\phi = \pi (1 - \cos \pi t / T) / 2$, for $t \leq T$ and $\phi = \pi$ for $t > T$, where $T = 0.4 \text{ s}$. To simulate an initial imperfection of the system, the tip of the beam is connected to the spherical joint via a rigid-body connection of length $d = 0.1 \text{ mm}$. The plane of the crank and link mechanism is offset from the plane of the beam by the same distance $d$.

The beam’s rectangular cross-section is of width $b = 10 \text{ mm}$ and height $h = 100 \text{ mm}$. The link
has a circular cross-section of radius $R_t = 12$ mm. Finally, the crank also features a circular cross-section, but its radius is $R_c = 24$ mm. All components are made of aluminum, whose mechanical characteristics are Young’s modulus $E = 73$ GPa and Poisson’s ratio $\nu = 0.3$. These physical properties translate to the sectional stiffness properties listed in table 3. The sectional mass properties are as follows: mass per unit span $m_{00} = 2.68$, 1.212, and 4.85 kg/m, moments of inertia per unit span $m_{22} = 2233$, 43.65, and 698.3, $m_{33} = 22.33$, 43.65, and 698.3 mg·m²/m for the beam, link, and crank, respectively.

<table>
<thead>
<tr>
<th></th>
<th>Axial</th>
<th>Shearing</th>
<th>Shearing</th>
<th>Torsional</th>
<th>Bending</th>
<th>Bending</th>
</tr>
</thead>
<tbody>
<tr>
<td>Beam</td>
<td>5.025</td>
<td>23.40</td>
<td>877.2</td>
<td>60,830</td>
<td>608.3</td>
<td></td>
</tr>
<tr>
<td>Link</td>
<td>10.81</td>
<td>10.81</td>
<td>914.5</td>
<td>1,189</td>
<td>1,189</td>
<td></td>
</tr>
<tr>
<td>Crank</td>
<td>43.22</td>
<td>43.22</td>
<td>14,630</td>
<td>19,020</td>
<td>19,020</td>
<td></td>
</tr>
</tbody>
</table>

4 General description of beam formulations

A beam is defined as a structure having one of its dimensions much larger than the other two, as depicted in fig. 4. The axis, or reference line, of the beam is defined along that longer dimension and its cross-section is normal to this axis. The cross-section’s geometric and physical properties are assumed to vary smoothly along the beam’s span.

Figure 4 depicts an initially curved and twisted beam of length $L$, with a cross-section of arbitrary shape and area $\mathcal{A}$. The volume of the beam is generated by sliding the cross-section along the reference line of the beam, which is defined by an arbitrary curve in space. Curvilinear coordinate $\alpha$ defines the intrinsic parameterization of this curve, i.e., it measures length along the beam’s reference line. Point $B$ is located at the intersection of the reference line with the plane of the cross-section.

4.1 Kinematics of the problem

In the reference configuration, an orthonormal basis, $\mathcal{B}_0(\alpha) = (\mathbf{b}_0, \mathbf{b}_2, \mathbf{b}_3)$, is defined at point $B$. Vector $\mathbf{b}_0$ is the unit tangent vector to the reference curve at that point, and unit vectors $\mathbf{b}_2$ and $\mathbf{b}_3$ define the plane to the cross-section. An inertial reference frame, $\mathcal{F}_1 = \{O, \mathcal{I} = (i_1, i_2, i_3)\}$, is defined, and the components of the rotation tensor that brings basis $\mathcal{I}$ to $\mathcal{B}_0$, resolved in basis $\mathcal{I}$, are denoted $R_{\mathcal{I}}(\alpha)$.

The position vector of point $B$ along the beam’s reference line is denoted $\mathbf{u}_0(\alpha)$. The position vector of material point $P$ of the beam then becomes $\mathbf{x}(\alpha_1, \alpha_2, \alpha_3) = \mathbf{u}_0(\alpha_1) + \alpha_2 \mathbf{b}_2 + \alpha_3 \mathbf{b}_3$, where $\alpha_2$ and $\alpha_3$ are the material coordinates along unit vectors $\mathbf{b}_2$ and $\mathbf{b}_3$, respectively. Coordinates $\alpha_1$, $\alpha_2$, and $\alpha_3$ form a natural choice of coordinates to represent the configuration of the beam.

In the deformed configuration, all material points located on a cross-section of the beam move to new positions. This motion is decomposed into two parts, a rigid-body motion and a warping displacement field. The rigid-body motion consists of a translation of the cross-section, characterized by displacement vector $\mathbf{u}(\alpha_1)$ of reference point $B$, and of a rotation of the cross-section, which brings basis $\mathcal{B}_0$ to $\mathcal{B}(\alpha_1) = (\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3)$, see fig. 4. The components of the position vector of point $B$ in the deformed configuration are denoted $\mathbf{x}(\alpha_1)$ and the components of the rotation tensor that brings basis $\mathcal{B}_0$ to $\mathcal{B}$ are denoted $R(\alpha_1)$; all tensor components are resolved in basis $\mathcal{I}$. 

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{beam_configuration.png}
\caption{Curved beam configuration.}
\end{figure}
For shear deformable beams, the deformation is characterized by six sectional strains: the axial strain, the two transverse shear strains, the twist rate, and the two bending curvatures.

4.2 Definition of the sectional strain

In the GEBF, the geometry of the beam is described by the displacement field of the beam’s reference axis and by the rotation field of its cross-section. The sectional strains and curvatures, denoted \( \varepsilon(\alpha_t) \) and \( \kappa(\alpha_t) \), respectively, are expressed [2] as

\[
\varepsilon(\alpha_t) = u_0 + u' - \left( R R^T \right) \hat{t}_1, \\
\kappa(\alpha_t) = \text{axial}(R R^T),
\]

where \( u_0 \) is the position vector of the beam’s axis in its reference configuration and notation \( (\cdot)' \) denotes a derivative with respect to \( \alpha_t \). The strain vector is defined as \( \varepsilon^T = \{ \varepsilon_{11}, \gamma_{12}, \gamma_{13} \} \), where \( \varepsilon_{11} \) is the sectional axial strain, and \( \gamma_{12} \) and \( \gamma_{13} \) the two components of transverse shearing strains.

4.3 The strain energy

The strain energy stored in the beam [19] of length \( L \) is expressed as

\[
A = \frac{1}{2} \int_0^L \varepsilon^{sT} \varepsilon^s \, d\alpha_t,
\]

where array \( \varepsilon^{sT} = \{ \varepsilon^s, \kappa^s \} \) stores the beam’s sectional strain components resolved in material basis \( \mathcal{B} \), i.e., \( \varepsilon^s = (R R^T)^T \varepsilon \) and \( \kappa^s = (R R^T)^T \kappa \). If the beam’s cross-section is made of isotropic materials and characterized by double symmetry, the sectional stiffness matrix referred to the centroid of the section is diagonal, \( \varepsilon^s = \text{diag}(S, K_{22}, K_{33}, H_{11}, H_{22}, H_{33}) \), where \( S \) is the beam’s axial stiffness, \( K_{22} \) and \( K_{33} \) its shear stiffnesses along unit vectors \( b_2 \) and \( b_3 \), respectively, \( H_{11} \) its torsional stiffness, and \( H_{22} \) and \( H_{33} \) its bending stiffnesses about unit vectors \( b_2 \) and \( b_3 \), respectively.

4.4 Equilibrium equations

The sectional forces and moments resolved in basis \( \mathcal{F} \) are denoted \( N \) and \( M \), respectively. When rotated to material basis \( \mathcal{B} \), the corresponding quantities are denoted \( N^s \) and \( M^s \), respectively where \( N^{sT} = \{ N^s_1, V^s_2, V^s_3 \} \) and \( M^{sT} = \{ M^s_1, M^s_2, M^s_3 \} \). The sectional axial force is \( N^s_1 \), and \( V^s_2 \) and \( V^s_3 \) are the shear forces along unit vector \( b_2 \) and \( b_3 \), respectively. Finally, \( M^s_1 \) is the torque, and \( M^s_2 \) and \( M^s_3 \) are the bending moments about unit vector \( b_2 \) and \( b_3 \), respectively. Array \( \varepsilon^{sT} = \{ \varepsilon^s, \kappa^s \} \) stores the six sectional stress resultants and the sectional constitutive laws are \( \varepsilon^s = \varepsilon^{sT} \varepsilon^s \).

Application of the principle of virtual work then yields the static equilibrium equations of the problem

\[
N' = -f, \\
M' + (\hat{b}_1 + u') \times N = -m.
\]

where \( f \) and \( m \) denote the externally applied forces and moments per unit span of the beam, respectively.

5 Description of the codes used in this effort

This paper will present the predictions of eight different codes for the four benchmark problems introduced in section 3. These eight codes are described in the present section in a very succinct manner. Table 4 lists the eight codes and the references providing details about them.

**Description of Dymore.** Dymore is based on the geometrically exact beam formulation. The sectional strains are defined by eq. (3) and the strain energy by eq. (4). The results presented in this
paper use a four noded element based on cubic Lagrangian shape functions. Each node features six degrees of freedom (DOFs), three displacement components and three rotation components. The Wiener-Milenković [21] parameters are used to represent finite rotations. A complete description of the formulations is found in refs. [20, 21].

**Description of Hotint.** HOTINT (http://www.hotint.org) is an open source multibody dynamics simulation code, which has been developed at the Johannes Kepler University Linz and at the Linz Center of Mechatronics. HOTINT includes various possibilities for the modeling of point masses, rigid bodies, beams, plates, and modally reduced multibody systems. The beam formulation is based on the ANCF described by Nachbagauer et al. [22]. The nodal coordinates are given by a displacement vector and two slope vectors, which coincide with the principal axes of the cross section in the reference configuration. As these two slope vectors are not necessarily perpendicular (but they are nearly) during deformation of the beam, a modification of constraints, moments and sensors which are related with the rotation of the cross-section is performed. Thus, for the computation of the rotation matrix from the two slope vectors in any point at the beam axis, a Gram-Schmidt projection of the slope vectors is utilized in order to obtain rotational parameters as provided in the results. The multibody system is represented by means of a differential algebraic equations of index 3, which is solved by means of a RadauIIA scheme with two stages (order 3) to provide numerical damping.

**Description of MBDyn.** The structural DOFs used by MBDyn are the absolute position of the nodes in reference $\mathcal{J}$ and the Cayley-Gibbs-Rodrigues [21] parameters, which are used to represent finite rotations. The approach uses an incremental scheme that resembles an updated Lagrangian approach, which has been termed *updated-updated* [23]. The constrained dynamics problem is formulated as DAEs that express Newton-Euler’s equations of motion of rigid bodies connected by deformable components and subjected to kinematic constraints in form of algebraic equations. It is integrated using an implicit, A/L stable multistep scheme with tunable algorithmic dissipation [23].

MBDyn supports GEBF elements implemented according to the finite-volume approach developed by Ghiringhelli et al. [24]. The generalized strains are defined according to eqs. (3). Integration by parts of the equilibrium equations (5), weighted by piecewise constant test functions centered on the nodes, yields the elastic contribution to the equilibrium of naturally discrete pieces of beam. When a three-node discretization is considered, with the interfaces between the mid- and the end-nodes at points corresponding to the two Gauss quadrature points, an intrinsically shear-locking free discrete element is obtained, which yields the exact static solution for any end-applied node. Inertia loads are accounted for using rigid-body elements at the nodes.

**Description of Mecano.** Mecano includes a large rotation nonlinear finite element beam based on the static equilibrium equations (5) resolved in the material frame [11, 25]. Rotations are parameterized using the rotational vector [21] and an updated Lagrangian scheme allows the handling of rotation in excess of $2\pi$ in magnitude. The element fields are interpolated linearly within the

---

**Table 4. Summary of the codes used for the numerical predictions.**

<table>
<thead>
<tr>
<th>Code name</th>
<th>Symbol</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dymore</td>
<td>*</td>
<td>[20, 21]</td>
</tr>
<tr>
<td>Hotint</td>
<td>○</td>
<td>[22]</td>
</tr>
<tr>
<td>MBDyn</td>
<td>△</td>
<td>[23, 24]</td>
</tr>
<tr>
<td>Mecano</td>
<td>▽</td>
<td>[11, 25]</td>
</tr>
<tr>
<td>MOPEDS</td>
<td>□</td>
<td>[26, 27]</td>
</tr>
<tr>
<td>Oofelie</td>
<td>×</td>
<td>[28, 29]</td>
</tr>
<tr>
<td>$SE(3)$</td>
<td>+</td>
<td>[30, 31]</td>
</tr>
<tr>
<td>Spacar</td>
<td>○</td>
<td>[32, 33]</td>
</tr>
</tbody>
</table>

---

1625
element and a single Gauss point is used for integration. To enable large finite rotations, incremental rotations with respect to the previous converged configuration are interpolated. This scheme requires tracking of total rotations both at the element nodes and at the Gauss point. This code, which is part of the LMS Samtech-SAMCEF software, has been used in industrial applications extensively for the last twenty years.

**Description of MOPEDS.** The beam finite element is based on the GEBF described in sections 4.1 to 4.4. The finite element formulation relies on standard Lagrangian shape functions used for the interpolation of the beam reference line, \( u_0(\alpha_1) + u(\alpha_1) \), and the rotation tensor, \( R(\alpha_1) \). The orthogonality of the rotation tensor is relaxed to the nodal points of the finite element. To this end, either Lagrangian multipliers or three (incremental) rotations are employed. Details of the finite element formulation can be found in Betsch and Steinmann [26]. The incorporation of the beam finite element formulation into a general framework for flexible multibody dynamics is described by Leyendecker et al. [27]. It is worth noting that the present approach makes possible the design of energy-momentum consistent time integrators. A refined version of the beam finite element formulation exhibiting improved convergence properties has recently been proposed by Eugster et al. [34]

**Description of Oofelie.** A large rotations nonlinear beam finite element model was developed by Lens and Cardona [28]. The finite element is able to handle large three-dimensional rotations and displacements but is restricted to small strains. Simplifications are made in the kinematics that allowed to obtain quite simple compact expressions. The kinematics expressions are based on assuming small relative displacements and rotations within the element. A single Gauss point is used to compute the strain energy and the expressions of rotations at the middle point are computed using a multiplicative decomposition of the increment of rotation from one node to the other. Then, curvatures and strains are obtained by assuming simple approximations for the derivative of the rotation tensor with respect to the axial coordinate, evaluated at the middle point. Full analytical expressions of the internal forces, inertia forces, stiffness, and mass matrices can be obtained, allowing an easy implementation of the element. The element was implemented in the finite element code Oofelie [29], which is being developed jointly between the University of Liège, the Universidad Nacional del Litoral and the company Open Engineering SA.

**Description of formulation in the Special Euclidean group.** The kinematics of mechanical systems are described using frame transformations, which are treated as elements of the special Euclidean group \( SE(3) \). Brüls et al. [35] used this approach to simulate rigid bodies. Generalizations to kinematic joints and beam elements were proposed by Sonneveld and Brüls [30] and Sonneveld et al. [31]. The constraint equations and the internal and inertia forces are naturally expressed in terms of unknowns evaluated in the material frame, and hence, the tangent matrix depends on local relative motions only and remains constant under rigid-body motions. By construction, the formulation is frame invariant. The equations of motion take the form of second-order differential-algebraic equations on a Lie group and are solved using a Lie group time integration scheme, the generalized-\( \alpha \) method proposed by Brüls et al. [35].

In the beam formulation, a material frame is attached to each point of the neutral axis of the beam, accounting for the position of the point and the orientation of the cross-section at that point with respect to the inertial frame. A two-node element is used and the finite element interpolation of the nodal frames is based on the exponential map of the special Euclidean group \( SE(3) \). The resulting shape functions are helicoidal functions and the coupling between rotations and translations they introduce yields a naturally locking-free element.

**Description of Spacar.** Spacar is based on the generalized strain beam formulation of Besselings [36] (http://www.utwente.nl/ctw/wa/software/spacar). A key point in this formulation is the selection of generalized strains as discrete deformations that are invariant for rigid-body motions of the element. The deformations are expressed as analytical functions of the absolute nodal coordinates in a co-rotational framework. The deformation functions include the specifica-
tion of rigid body motions as displacements for which the discrete deformations are zero. This avoids the shortcoming of standard co-rotational formulations when describing rigid bodies. Flexible elements are handled by allowing non-zero deformations and specifying constitutive relations between the discrete deformations and stress-resultants. The derivation of the element stiffness matrix is based on a discretization of the elastic line of a three-dimensional Timoshenko beam model in a co-rotational frame, whereas the inertia properties are derived using a discretization of the elastic line in the inertial frame of reference.

Geometric non-linearities arising from changes in geometry involving finite deflections and pre- and post-buckling, are approximated by additional second-order terms in the expressions for the deformations. These second-order terms are derived from a non-linear continuum model of the elastic line in the co-rotational frame, including a finite strain description proposed. A Taylor series expansion is used to express the non-linear curvature and strain displacement equations into a second-order polynomial. Integrating these equations over the length of the beam and using the second moment-area theorem yields the additional second-order terms describing the geometric couplings among the axial elongation, bending, and torsion deformations. A detailed description of the formulation is found in Jonker et al. [32, 33].

6 Comparison of numerical predictions

In this section, the numerical predictions of the beam formulations described in sections 4 and 5 will be compared for the four benchmark problems listed in section 3. The first and second columns of table 4 lists the name of the formulations and the symbol that will be used to identify the numerical predictions of each codes.

6.1 The Princeton beam experiment

In this correlation effort, the distributed weight of the beam was neglected. Unit vector \( \bar{e}_3 \), see fig. 1, was left in the vertical orientation and the orientation of the tip load was rotated from 0 to 90 degrees. While this represents an approximation, its effect on the numerical predictions is far smaller than the observed scatter in the experimental measurements.

Figures 5, 6, and 7 show the tip flapwise displacement, \( u_2 \), chordwise displacement, \( u_3 \), and twist, \( \phi \), respectively. The experimental measurements (average \( \circ \) and error bars) for each of the three loading cases, labeled \( P_1 \), \( P_2 \), and \( P_3 \), respectively, are shown in the figures. The numerical predictions of the eight codes presented in section 5 are also shown, using the symbols listed in table 4. For reference, the linear solution given by eqs. (2) is also shown in figs. 5 and 6 with dotted lines; the linear solution predicts a vanishing tip twist. The tip rotation is the quantity most affected by nonlinear behavior. The eight predictions of the maximum values of the tip rotation for load case \( P_3 \) were averaged to find \( \mu_\phi = 0.06177 \) rad and the coefficient of variation was \( \sigma_\phi/\mu_\phi = 0.0076 \), where \( \sigma_\phi \) is the standard deviation of the distribution. Clearly, the predictions of the eight codes are in very close agreement.

Note that the Dowell and Traybar report [17] provides no measurements for loading condition \( P_2 \) at loading angles \( \theta = 75 \) and 90 degrees and for loading condition \( P_3 \) at loading angles \( \theta = 60, 75, \) and 90 degrees. A cursory look at fig. 5 reveals that those loading cases would result in large flapwise deflections, which could generate permanent plastic deformations in the beam. It is likely that the authors of the study did not want to damage the test article and hence, did not acquire data at those loading conditions.

6.2 The four-bar mechanism

For this problem, the simulation was run for three complete revolutions of the crank, starting from initial conditions at rest. At the beginning of the first revolution, high-frequency oscillations are observed, but due to the algorithmic dissipation, these oscillations have all but disappeared in the second revolution. The results presented in the sequel are the numerical predictions for the third
revolution. This problem was simulated for a total of 12 s using 3000 time steps of constant size \( \Delta t = 4 \) ms.

The components of axial force, \( F_1 \), and bending moment, \( M_2 \), along unit vector \( \hat{b}_1 \) and about unit vector \( \hat{b}_2 \), respectively, at the mid-span of bar 1, both resolved in the material basis are depicted in figs. 8 and 9, respectively. The component of rotation of bar 2 at point C, \( \phi \), is shown in fig. 10. At point C, the Euler angles (sequence 3-1-2) defining the orientation of bar 2 in the inertial basis are computed and the first angle of the sequence is presented in the figure. The relative rotation, \( \theta \), at the revolute joint at point D is depicted in fig. 11. The predictions of the eight codes are shown using the symbols listed in table 4.

The four figures show that the predictions of the eight codes are in excellent agreement with each other. To quantify the quality of the agreement, the eight predictions of the minimum values of the axial force shown in fig. 8 were averaged to find \( \mu_{F_1} = 5.966 \) N and the coefficient of variation was \( \sigma_{F_1}/\mu_{F_1} = 0.0043 \), where \( \sigma_{F_1} \) is the standard deviation of the distribution. Clearly, the predictions of the eight codes are in very close agreement. Similarly, the average of the eight maximum relative rotations at point D appearing in fig. 11 was found to be \( \mu_{\theta} = 1.579 \) rad and the corresponding coefficient of variation was \( \sigma_{\theta}/\mu_{\theta} = 0.0032 \).
7 Conclusions

This paper has described in details four benchmark problems for the validation of beam models used in flexible multibody dynamics simulations. For each of the four benchmark problems, the numerical predictions of eight independent codes have been presented and the formulations on which these codes are based have been described succinctly.

Many of the formulations are related closely to the geometrically exact beam formulation but they differ in their finite element implementation, time integration scheme, treatment of the constraints and invariants, and solution techniques. The eight formulations have been implemented by eight researchers independently. Yet, this paper shows that the predictions obtained by the eight researchers are in very good agreement with each other and with experimental results, when available.

Because the proposed benchmark problems have been described accurately and have been run by eight different researchers successfully, they are expected to be useful to other researchers for the validation of future beam element formulations. All the numerical predictions presented in the paper are available in electronic format at http://www.dymoreolutions.com/Benchmarks/Benchmarks.html.

The present paper has focused on the consistency of the predictions of eight different codes. The efficiency of these codes, however, has not been assessed. Indeed, assessment of computation efficiency would require the eight codes to be run on the same computer hardware, or at least to evaluate their computational complexity. This difficult issue will have to be addressed in the future to be able to assess both accuracy and efficiency of the various formulations.

REFERENCES


AUTHORS INDEX
This volume contains the full papers of the ECCOMAS Thematic Conference on Multibody Dynamics 2015 held in Barcelona on June 29 - July 2, 2015.

Barcelona School of Industrial Engineering (ETSEIB)
Department of Mechanical Engineering
Universitat Politècnica de Catalunya
Diagonal 647, 08028 Barcelona
Catalonia, Spain
multibody2015.org