11th World Congress on Computational Mechanics (WCCM XI)
5th European Conference on Computational Mechanics (ECCM V)
6th European Conference on Computational Fluid Dynamics (ECFD VI)
E. Oñate, J. Oliver and A. Huerta (Eds)

DEVELOPMENT OF A NEW CALCULATION SOFTWARE FOR LARGE DEFORMATION PROBLEMS

V. SATTINGER[*], D. SUPANZ[†] AND I. TURCIN[†]

[*] CAMPUS 02 University of Applied Sciences
Degree Programs in Automation Technology
Körblergasse 126, A-8010 Graz, Austria
e-mail: vinzenz.sattinger@campus02.at, www.campus02.at

[†] CAMPUS 02 University of Applied Sciences
Degree Programs in Automation Technology
Körblergasse 126, A-8010 Graz, Austria
e-mail: ioan.turcin@campus02.at, www.campus02.at

Key Words: *FEM-Software development, Parallel algorithms, Finite Element technology, Solid-Shell Elements, Hyperelasto-plastic deformation, Locking phenomena.*

Abstract. This paper covers the development of a new stand-alone 3D finite element software package called "MSIM", which allows large deformation calculations as it occurs in the field of manufacturing technology. Since existing commercial software packages do need a lot of time to solve the highly nonlinear system of equations and do not often incorporate suitable elements as it is common in several forming processes, a new calculation program was implemented to remedy this deficiency. The first part of this work describes the development and the object-oriented implementation of the finite element program MSIM in C++, which is highly superior to commercial software packages with respect to calculation time and storage capacities. The parallelization of the assembly algorithm and the implementation of the solution algorithm for the highly nonlinear system of equations is one of the core issues. The second part focuses on the development of a novel solid-shell element and treats the locking phenomenon resulting from low order interpolation functions. The customized solid-shell element formulation is embedded in a static implicit total Lagrange formulation, which is able to deal with large deformations in accordance with the finite strain theory. For the constitutive relation, a hyperelastic material law was used in preparation to a further extension to a hyperelastic-plastic material behavior. Therefore, the determination of the elastic predictor gained from the right Cauchy Green strains, turned out to be a crucial step in hyperelasto-plasticity. Several benchmark tests were performed to evaluate the computing speed of the new calculation program MSIM and the accuracy of these solid-shell elements in comparison to the results attained by the FEM-package ANSYS©. The results obtained so far agrees satisfactorily by gaining a significantly reduced calculation time. The developed calculation program serves as an effective tool to predict the stress-distributions inside the material under large deformations. Furthermore, this calculation program provides the basis for further improvements and enhancements and thus serves as an effective tool for research and teaching.

1 INTRODUCTION

Simulations in the field of manufacturing technology, which requires to take into account large deformations, large kinematic movements (esp. large rotations) and/or contact and friction, leads to highly nonlinear system of equations and finally to a time-consuming solving procedure. Well known and established commercial finite element software packages like ANSYS©, ABAQUS© a.s.o, are very expensive to acquire for small specialized companies. These software packages are equipped with plenty of standard finite element formulations, which are often not suitable in the presence of large deformations, temperature and/or phase transitions as it is common in metal forming processes. Therefore, these software packages provide programmable interfaces to establish user defined element formulations or material laws (e.g. UMAT-subroutine in ABAQUS©), where the user have the opportunity to implement his own program code. Another possibility is to use open-source software packages (e.g. Z88, FreeFEM++), but they are often not well documented and may contain undiscovered bugs. Therefore, whether commercial software packages or open-source software packages, the user have to implement his own program code at special application cases. Another aspect arises from the large calculation time caused by the general formulation of the contact search algorithm, which can be reduced by using a problem adapted contact algorithm [6, 12].

These arguments resulted in the decision to develop a customized calculation software prototype. The advantage is clear: no license fees under a high degree of flexibility for special application problems.

2 OBJECTIVES

The aim of this study is to develop a finite-element calculation program, taking into account:

- Large deformations as it is present in many metal-forming processes.
- Flexibility in terms of further element enhancements.
- Efficiency with regard to a low solving time.

Based on these arguments the calculation program is built in C++ by using solid-shell elements, because these elements take into account the full three dimensional stress-state as they are needed in the field of metal forming processes. Many authors gives instructions respectively a schematic flowchart to build a finite-element or a multibody calculation program e.g. [1, 3, 8, 14], which will serve as a guideline in this work.

3 SOFTWARE DEVELOPMENT

The software development has been carried out in an object-oriented way in Microsoft Visual Studio C++. The class hierarchy is shown in Figure 1. The base classes CPart and CElement provides the basic functionality for all further derived classes. Each new derived part or element inherits the main functionality of his base class and allows a simple enhancement of new properties. The class CAssembly is used as a container which includes the assembly as well as the equation solving algorithm. CExtList and CMatProp contain all the functionality and information regarding the boundary conditions and material properties. Large deformation problems leads generally to the static-implicit equation system

$$\mathbf{R}(\mathbf{u}) = \mathbf{P} \tag{1}$$

where $\mathbf{R}(\mathbf{u})$ describes the vector of inner forces and $\mathbf{P}$ describes the Vektor of external forces. Equation 1 is solved by the Newton-Raphson procedure. Therefore, the linearized equation system

$$\mathbf{K}\,\Delta\mathbf{u} = \Delta\mathbf{P} \tag{2}$$

with the global tangent stiffness matrix $\mathbf{K}$ is the basis for an iterative solving procedure in CAssembly, which will be solved by a parallelized sparse-solver (pardiso-solver) as provided by the Intel©-MKL-Library [15, 16]. The sparse-matrix storage scheme is also shown in [15, 16] and implemented in the class CMatrixSparse.

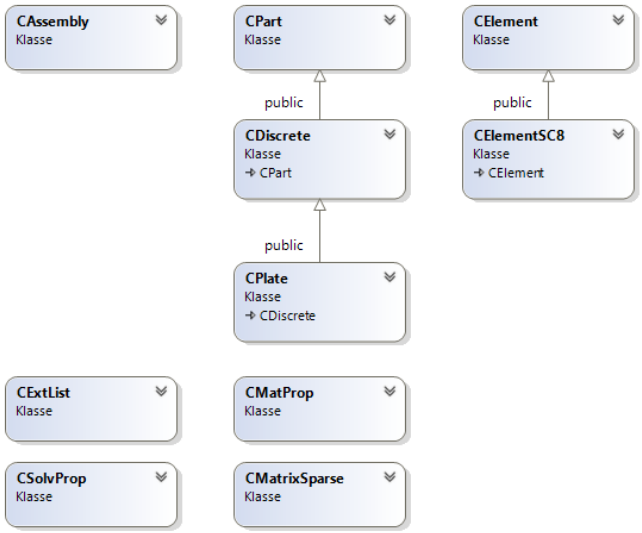

Figure 1: Class hierarchy in C++ of the calculation program MSIM

3.1 Parallelization algorithm

The assembly operation is parallelized to keep the total solving time low and exploit the power of multicore processors. For the assembly operation, it is always necessary to use a loop over all affected elements and therefore, a linear list is established (see Figure 2). The linear list including all elements are split into several sections, whereas each section will be executed by one thread.

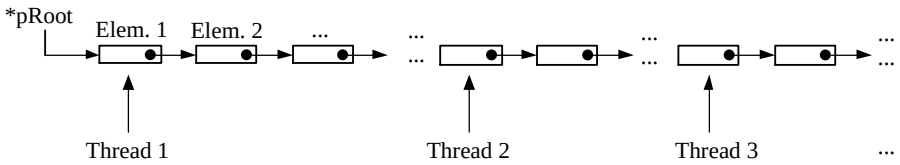

Figure 2: Parallelization of the element list

Thread 1 begins at the top of the list and assembles all elements in the direction of the start point of thread 2. The next section is executed in a similar way by thread 2 and so on. The total count of threads will be automatically determined by the calculation software and is depending on the size of the element list and the computer being used. This parallelization technique turned out to be very efficient and led to a significantly speed-up in calculation time especially at high order equation system.

4 ELEMENT DEVELOPMENT

In the following paragraph a new solid shell element (denoted by CElementSC8 in Figure 2) will be developed by derivation from the base class CElement, which is used to calculate large elastic deformations (e.g. rubber, gaskets a.s.o.). Therefore, a hyperelastic material law by an underlying isotropic strain energy functional

$$W(J,\hat{\mathbf{C}}) = U(J) + \hat{W}(\hat{\mathbf{C}}) \tag{3}$$

consisting of the volumetric part

$$U(J) = \frac{\kappa}{4}\left[\left(J-1\right)^2 + \left(\ln J\right)^2 \right] \tag{4}$$

and the deviatoric part

$$\hat{W}(\hat{\mathbf{C}}) = \frac{\mu}{2}\left(I_{\hat{\mathbf{C}}} - 3\right) \tag{5}$$

defined in the principal strains λ_i, with the relationships

$$J = \lambda_1 \lambda_2 \lambda_3 \tag{6}$$

$$I_{\hat{\mathbf{C}}} = \hat{\lambda}_1^2 + \hat{\lambda}_2^2 + \hat{\lambda}_3^2 \tag{7}$$

$$\hat{\lambda}_i = J^{-1/3} \lambda_i \tag{8}$$

will be used [4, 5, 11, 13]. Herein, J describes the volumetric part of the right Cauchy-Green strains $\mathbf{C}$ and $I_{\hat{\mathbf{C}}}$ denotes the first invariant of the deviatoric Cauchy-Green strains $\hat{\mathbf{C}}$.

4.1 Discretization of the solid shell element

Starting point for the solid shell discretization is the weak form

$$\int_{\Omega_0} \mathbf{S} : \delta\mathbf{E}\, dV = \int_{\Omega_0} \rho_0 \mathbf{b} \cdot \delta\mathbf{u}\, dV + \int_{\partial\Omega_0} \mathbf{t}_0 \cdot \delta\mathbf{u}\, dA \tag{9}$$
$$with \;\; \delta\mathbf{u} = 0 \;\; on \;\; \partial_u\Omega_0 \quad \forall\, \delta\mathbf{u}$$

in the Total-Lagrangian form, which includes the 2nd Piola-Kirchhoff stresses $\mathbf{S}$, the Green-Lagrangian stresses $\mathbf{E}$, the mass density ρ_0 referred to the initial configuration, the volume force vector $\mathbf{b}$ and the surface stress vector $\mathbf{t}_0$ acting on the initial configuration. For the

discretization of an eight-node solid shell element based on the degenerative concept, linear shape functions

$$N_K = 0{,}125\left(1+\theta_1^K\theta_1\right)\left(1+\theta_2^K\theta_2\right)\left(1+\theta_3^K\theta_3\right) \qquad K = 1...8, \tag{10}$$

were used (see Figure 3). Similar approaches can be found in [2, 7, 10].

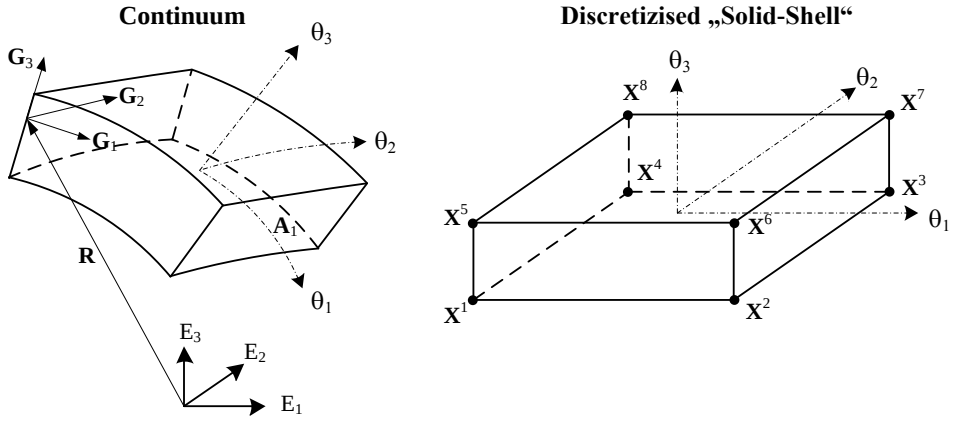

Figure 3: Kinematics and discretization of an 8-node solid shell element

An arbitrary point at the initial (Lagrangian $\mathbf{R}$) and current (Eulerian $\mathbf{r}$) configuration is expressed by the position vectors

$$\mathbf{R} = N_I\,\mathbf{X}^I \tag{11}$$

$$\mathbf{r} = N_I\,\mathbf{x}^I \tag{12}$$

Furthermore, all covariant basis vectors are given by

$$\mathbf{G}_i = \frac{\partial \mathbf{R}}{\partial \theta_i} \qquad \text{and} \qquad \mathbf{g}_i = \frac{\partial \mathbf{r}}{\partial \theta_i}. \tag{13}$$

Now the covariant components of the Green-Lagrangian strains and its variation

$$\bar{E}_{ij} = \frac{1}{2}\left(\mathbf{g}_i \cdot \mathbf{g}_j - \mathbf{G}_i \cdot \mathbf{G}_j\right) \tag{14}$$

$$\delta E_{ij} = \frac{1}{2}\left(\mathbf{g}_i \cdot \delta\mathbf{g}_j - \delta\mathbf{g}_i \cdot \mathbf{g}_j\right) \tag{15}$$

can be determined.

4.2 ANS method

The ANS method will be used to avoid the shear and thickness locking phenomena, which have a negative effect on the element response especially in bending dominated applications. For this purpose, special evaluation points on A, B, C and D for calculating the shear strains

$\bar{E}_{23}$ and $\bar{E}_{13}$, and E, F, G, H for calculating the thickness strains $\bar{E}_{33}$, are selected and subsequently interpolated over the continuum (see Figure 4).

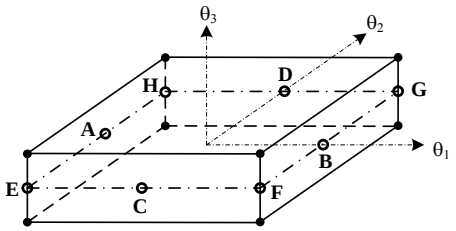

Figure 4: Evaluation points of the ANS method

This leads to the new shear strains:

$$\bar{E}_{23}^{ANS} = 0,5\left(1-\theta_1\right)\bar{E}_{23}^{A} + 0,5\left(1+\theta_1\right)\bar{E}_{23}^{B} \tag{16}$$

$$\bar{E}_{13}^{ANS} = 0,5\left(1-\theta_2\right)\bar{E}_{13}^{C} + 0,5\left(1+\theta_2\right)\bar{E}_{13}^{D} \tag{17}$$

$$\bar{E}_{33}^{ANS} = N_E\,\bar{E}_{33}^{E} + N_F\,\bar{E}_{33}^{F} + N_G\,\bar{E}_{33}^{G} + N_H\,\bar{E}_{33}^{H} \tag{18}$$

$$\text{with } N_K = 0,25\left(1+\theta_1^K\theta_1\right)\left(1+\theta_2^K\theta_2\right) \qquad K = E,F,G,H.$$

The shear and thickness strains $\bar{E}_{23}^{ANS}$, $\bar{E}_{13}^{ANS}$ and $\bar{E}_{33}^{ANS}$ obtained in this way, substitute the strains $\bar{E}_{23}$, $\bar{E}_{13}$ and $\bar{E}_{33}$ in Equation 14.

The 2nd Piola Kirchhoff stresses are calculated from the constitutive relation in Equation 3 based on the spectral decomposition

$$\mathbf{S} = \sum_{i=1}^{3} S_i\,\mathbf{N}_i \otimes \mathbf{N}_i\,. \tag{19}$$

The contravariant components $\bar{S}^{ij}$ can be calculated after some transformations. In preparation of a hyperelastic-plastic material law and taking into account the Levi-Mises flow rule with an underlying predictor-corrector algorithm, the elastic predictor related to the initial configuration leads to

$$\mathbf{C}_{n+1}^{e,Trial} = \mathbf{C}_{n}^{p\,-1}\,\mathbf{C}_{n+1}\,. \tag{20}$$

The crucial step is that the right Cauchy-Green strains $\mathbf{C}_{n+1}$ has to be calculated from Equation 20, which includes all relevant modifications caused by the ANS method. Since $\mathbf{C}_{n}^{p-1}$ and $\mathbf{C}_{n+1}$ is symmetric, the elastic predictor $\mathbf{C}_{n+1}^{e,Trial}$ is unsymmetrical. Therefore, a Cholesky decomposition with $\mathbf{C}_{n+1} = \mathbf{L}_{n+1}\,\mathbf{L}_{n+1}^{T}$ is applied, which leads to the equation

$$\underbrace{\mathbf{L}_{n+1}^{T}\mathbf{C}_{n}^{p-1}\mathbf{L}_{n+1}}_{\mathbf{A}}\underbrace{\mathbf{L}_{n+1}^{T}\mathbf{N}_{i}}_{\mathbf{y}_i}=\lambda_{i}^{e,Trial}\underbrace{\mathbf{L}_{n+1}^{T}\mathbf{N}_{i}}_{\mathbf{y}_i}, \tag{21}$$

by using the lower triangle matrix $\mathbf{L}_{n+1}$. The principal strains $\lambda_i^{e,Trial}$ and the directions $\mathbf{N}_i$ can be calculated by using the Jacobi Iteration on $\mathbf{A}$ and further substituted in the strain energy function (Equation 3).

5 ELEMENT TESTS

In the following the performance of the calculation program MSIM equipped with the solid shell element formulation will be demonstrated in contrast to the commercial software package ANSYS©. Figure 5 shows a clamped beam for which the element response was tested in the static implicit case for elastic material.

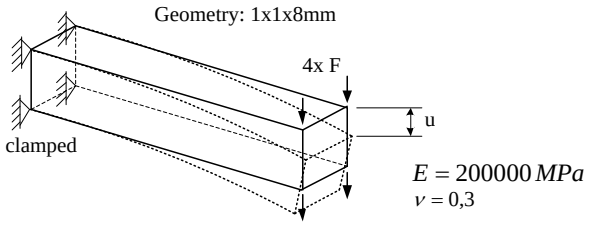

Figure 5: Deflection of a clamped beam

For this purpose different meshes were generated in the commercial software package ANSYS© and applied to the beam. The vertical tip deflection u after applying the external forces F (Figure 7) is very close to the ANSYS© solution, confirming the correct response of the SC8-ANS solid shell element. Table 1 shows the result and demonstrate a very good approximation even for low element numbers involved.

Table 1: Elastic response of the solid shell element

ANSYS©	**ANSYS©**	**MSIM (C++)**
3x1x23 elements SHELL 181 u = 1,58 mm	5x5x40 elements SOLID 185 u = 1,59 mm	1x1x8 elements SC8-ANS u = 1,55

Figure 6 shows a supported plate with one edge clamped.

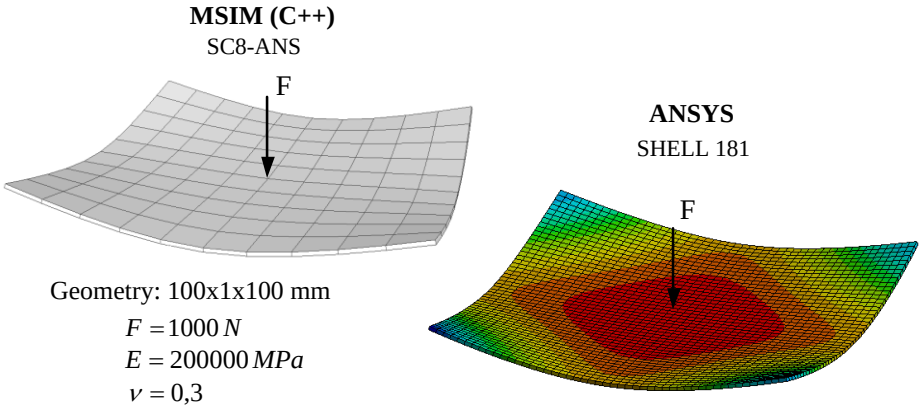

Figure 6: Deflection of a supported plate.

The corresponding vertical deflection u in the middle of the plate after applying the external force F is presented in Figure 7 at different load steps. Whereas at load step four, the total force is applied.

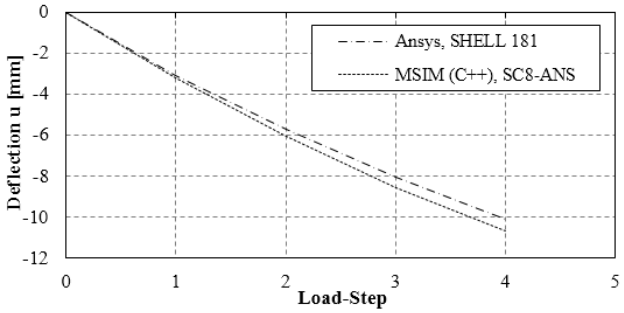

Figure 7: Deflection of a clamped plate depending on the load step.

The results show also a very good approximation in opposite to the solution gained from the commercial software package ANSYS© by using SHELL 181 elements. The great advantage lies in the significantly reduced calculation time of MSIM. MSIM needs 2 seconds which is superior to 50 seconds needed by ANSYS©.

The solid shell element formulation SC8-ANS also turns out to be very robust at large deformations as shown in Figure 8.

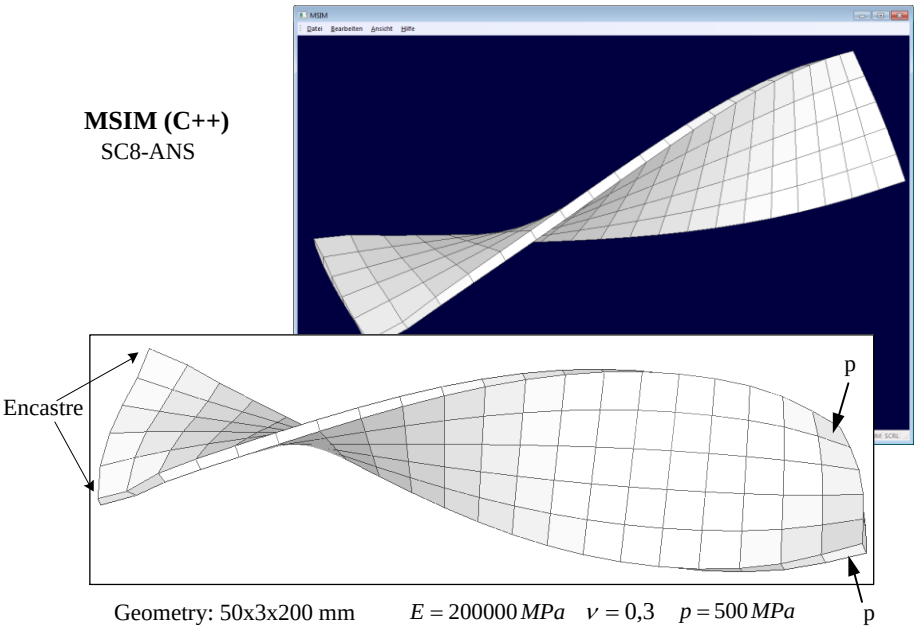

Figure 8: Plate under large torsional load

By using the same discretization and SHELL 181 elements in ANSYS©, the calculation failed after some load steps.

Beyond this, a lot of tests have been performed to evaluate the element behaviour at different load conditions. All of them turned out to be in satisfactory agreement with the ANSYS© solutions.

6 CONCLUSIONS

The self-developed calculation program MSIM turned out to be very efficient regarding calculation time in comparison with the commercial software package ANSYS© and fulfills the requirements posed at the beginning of this paper. The object oriented software development provides the flexibility to implement further elements and contact formulations. The parallelization of the assembly algorithm combined with the parallelized sparse-solver "pardiso" turned out to be very efficient.

Several systematic element tests beyond the tests mentioned above, confirm the robustness and convergence in the hyperelastic case and provides the basis for a hyperelasto-plastic extension [9].

REFERENCES

[1] Bathe K.J.: *Finite Element Procedures*, Prentice Hall, New Jersey, (1996).

[2] Bischoff M.: *Theorie und Numerik einer dreidimensionalen Schalenelement-formulierung*, Institut for Structural Mechanics, University of Stuttgart, Technical report, (1999).

[3] Crisfield M.A.: *Non-linear Finite Element Analysis of Solids and Structures*, Volume 1,

John Wiley & Sons, (1996).
[4] Choi K., Im S.: Finite element simulation of welding processes using a solid-shell element, *Journal of Physics* (2009) **42**:1-17.
[5] Masud A., Tham C.L., Liu W.K.: A stabilized 3-D co-rotational formulation for geometrically nonlinear analysis of multi-layered composite shells, *Comput. Methods. Appl. Mech. Eng.* (2000) **26**:1-12.
[6] Sattinger V., Kainz A., Schörkhuber K., Aigner L., Zeman K.: Calculation of mechanical and thermal influences during coiling of hot strip, *Comput. Plasticity XII – Fundamentals and Applications* (2013) **1**:1186-1197.
[7] Schwarze M., Vladimirov N., Reese S.: Sheet metal forming and springback simulation by means of a new reduced integration solid-shell finite element technology, *Comput. Methods Appl. Mech. Eng.* (2011) **200**:454-476.
[8] Shabana A.A.: *Dynamics of Multibody Systems*, Cambridge University Press, (2010).
[9] Simo J.C., Miehe J.C.: Associative coupled thermoplasticity at finite strains: formulation numerical analysis and implementation, *Comput. Methods Appl. Mech. Eng.* (1992) **68**:1-31.
[10] Sousa R.J.A., Cardoso R.P.R., Valente R.A.F., Yoon J.W., Gracio J.J., Jorge R.M.N.: A new one-point quadrature enhanced assumed strain (EAS) solid-shell element with multiple integration points along thickness, Part II: Nonlinear applications, *Int. Journal for Num. Meth. Eng*. (2006) **67**:160-188.
[11] Wriggers P.: *Nichtlineare Finite-Element-Methoden*, Springer, (2000).
[12] Wriggers P.: *Computational Contact Mechanics, Second Edition*, Springer, (2006).
[13] Zhongqing Zhou, Thomson P.F., Yee Cheong Lam, Yuen D.W.: Numerical analysis of residual stress in hot-rolled steel strip on the run-out table, *Journal of Materials Processing Technology* (2003) **132**:184-197.
[14] O.C. Zienkiewicz and R.C. Taylor, *The finite element method*, 6th Edition, Elsevier, (2005).
[15] Intel Math Kernel Library: *Reference Manual*, September (2007).
[16] Intel Math Kernel Library: *User's Guide - Intel Math Kernel Library*, Version 10.3.