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)

CODE VERIFICATION OF A PARTITIONED FSI ENVIRONMENT FOR WIND ENGINEERING APPLICATIONS USING THE METHOD OF MANUFACTURED SOLUTIONS

RUPERT FISCH*, JÖRG FRANKE†, ROLAND WÜCHNER* AND KAI-UWE BLETZINGER*

*Chair of Structural Analysis
Technische Universität München (TUM)
Arcisstraße 21, 80333 Munich, Germany
e-mail: rupert.fisch@tum.de, web page: www.st.bgu.tum.de

† Vietnamese-German University (VGU)
Le Lai Street, Hoa Phu Ward, Binh Duong New City, Binh Duong Province, Vietnam
e-mail: joerg.franke@vgu.edu.vn, Web page: www.vgu.edu.vn

Key words: code verification, benchmarks, method of manufactured solutions, partitioned FSI, non-matching grids

Abstract. This contribution presents an effective method to perform Code Verification of a partitioned software environment which is designed for Fluid-Structure Interaction (FSI) problems. The main focus lies on elastic membrane structures in an unsteady flow regime. Code Verification is a part of efforts to guarantee the code's correctness and to obtain finally predictive capability of the code [15]. The Method of Manufactured Solutions turned out to be an effective tool to perform Code Verification, especially for complex physics and partitioned approaches. The given examples provide a hierarchical benchmark suite for the reader to assess other codes, too. The given benchmarks deal with a fully coupled structural dynamics software $Carat++$ with the fluid software $OpenFOAM^{®}$. The Dirichlet-Neumann coupling is performed by the software *Empire*. The whole partitioned FSI environment is assessed to converge as intended. Especially the topic of non-matching grids at the common interface is assessed. Once the confidence in the general environment of the numerical wind tunnel is established, one will be able to discuss specific issues that surpass the limitations of the experimental wind tunnel.

1 INTRODUCTION

Within this work, the solution quality and hence the prerequisite for predictive capability of a partitioned Fluid-Structure Interaction (FSI) simulation environment for

lightweight structures (e.g. fig. 1 right) is assessed. These structures are highly sensitive to dynamic loading, for instance induced by wind. The increase in computational power leads engineers more and more towards computer aided approaches in wind engineering as a complement to the experimental wind tunnel. Within the design of a numerical wind tunnel, one has to care about a lot of aspects, like turbulence, modelling of the atmospheric boundary layer, creating geometric models, solving the problem with all relevant influencing physical factors. Besides that, the justified skepticism in new approaches has been and still is very high. Thus, only serious work on the predictive capability of Computational Wind Engineering (CWE) can generate trust for this approach.

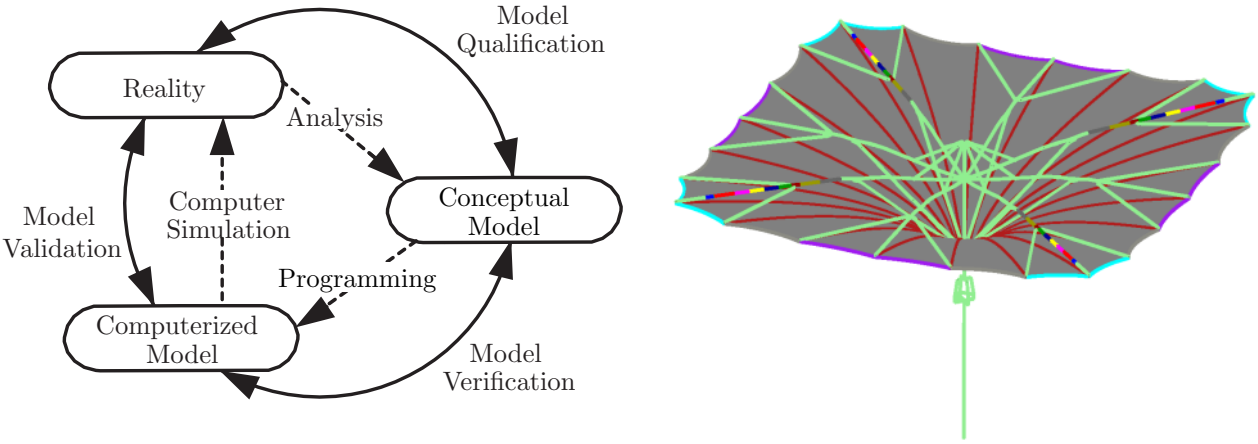


Figure 1: Phases of Modeling & Simulation and the role of V&V (left) [20]; A lightweight umbrella structure from a collaboration with SL Rasch, http://www.sl-rasch.de (right).

2 CODE VERIFICATION

Oberkampf and Roy name a list of influence factors for the predictive capability of simulations [15]. One factor is Verification and Validation (V&V). These provide evidence for the correctness of the code as well as the results and contain a concept of quantitative accuracy assessment (cf. figure 1 left). Verification means the "process of assessing software correctness and numerical accuracy of the solution to a given mathematical model" [13], or simpler spoken "Verification is about solving the equations right" [2]. Validation means the "process of assessing the physical accuracy of the mathematical model based on comparison between computational results and experimental data" [15], or simpler spoken "Validation is about solving the right equations" [2]. One part of verification is called code verification. It "ensures that the underlying mathematical model, e.g. a Partial Differential Equation (PDE) is correctly implemented in the computer code and identifies software errors" [15].

2.1 Code Verification

To perform code verification, order-of-accuracy tests are highly recommended [14, 19]. They are the most meaningful tests to assess a code's correctness.

2.2 The Method of Manufactured Solutions

For Code Verification, the Method of Manufactured Solutions (MMS) has been used in fluid dynamics, e.g. in [3, 11, 15, 18, 19], but also in structural stress computations [12] or recently in monolithic fluid-structure interaction computations [4]. The MMS can be used as a toolbox to assess implemented numerical procedures for the solution of partial differential equations (PDE). The principle idea of the MMS is to build own analytical solutions. The analytical (manufactured) solution is in general not fulfilling the actual PDE. Thus, one has to add source, resp. force, terms to the discretized algebraic equations, such that the program is able to compute the manufactured solution asymptotically.

If all numerical schemes are correctly implemented one can observe that the solution $\mathbf{b}$ tends towards the manufactured solution $\hat{\mathbf{b}}$ for systematically refined calculations. As $\hat{\mathbf{b}}$ states the exact solution, the difference between $\mathbf{b}$ and $\hat{\mathbf{b}}$ represents the exact error of the individual calculation. The development of the error with refinement (refinement factor $r = h_{coarse}/h_{fine}$ with a characteristic element and/or timestep size h) gives the observed order of accuracy p in equation (1) [11].

$$p = \frac{\log\left(\frac{E_{h_{coarse}}}{E_{h_{fine}}}\right)}{\log\left(r\right)} \tag{1}$$

If the formal order of accuracy $\hat{p}$ (cf. section 2.3 on the following page and section 3.4 on page 6) matches p in the asymptotic range of the solution, the following parts of the code have been verified [17]: all coordinate transformations, the order and the programming of the discretization, and the matrix solution procedure. If the two orders p and $\hat{p}$ do not match, there can be many reasons, e.g. programming mistakes, insufficient grid resolution, singularities, etc. An extensive discussion of reasons is given in [15]. The used MMS procedure with all its parts is shown in detail in section 4 on page 6.

The limitations of the MMS are discussed in the literature (e.g. [17]). The prerequisites for the application of the MMS can be found in detail e.g. in [19]. The basic requirement is that the solution $\hat{\mathbf{b}}$ represents a smooth analytical function with a sufficient number of derivatives in the domain space. Furthermore, $\hat{\mathbf{b}}$ and the physical parameters of the PDE should be general enough and well balanced to activate all terms of the governing equations. The independence of the numerical approach turned out as a major advantage of the MMS: The equations of state are handled in the continuum, therefore it is completely independent of the discretization method (e.g. Finite Elements, Finite Volume,...) or the solution procedure (direct solution, fix-point iteration, Newton-Raphson,...). This point made it very attractive for the assessment of the existing partitioned FSI framework, which consists of different programs and different discretization methods. In this paper

the given solutions give a stair of complexity or even a benchmark series to assess all containing parts of an partitioned FSI process step by step. This stair can be very helpful to localize coding or other mistakes [11, 19]. Therefore each solution should be as simple as possible but as complex as necessary. The MMS is applied to assess the Finite Element formulation of a fully geometric nonlinear membrane element in an unsteady regime, fully coupled with the Finite Volume formulation of unsteady incompressible fluid dynamics.

2.3 Formal Order of Accuracy

The discretization method, the chosen form functions for spatial and temporal discretization, and the solution process determine the formal order of convergence. One can determine the formal order of convergence $\hat{p}$ of an approximated expression by comparison with its Taylor series expansion. The first term of the Taylor series, which is not approximated, represents the main error in the asymptotic range of the solution. In general, the derived formal order can be observed as long as no other errors (e.g. geometry, integral, solution approximation) with smaller convergence rate occur. A detailed discussion can be found in [1, 8, 10, 22, 24].

2.4 Discrete Field Error

In this paper, we concentrate on the mathematical exercise and the assessment of the code's correctness. Therefore the error, meaning the difference between the exact solution $\hat{\mathbf{b}}$ and the approximated solution $\mathbf{b}$ running the program, contains discretization, round-off, and IICE error [11]. As the round-off and the IICE are kept very low, the leading error is the discretization error as long as there are no programming mistakes. To determine the global error of a field $\mathbf{b}$ in the domain K one can use error norms. The L2 norm is chosen to have a representative mean field error of the complete domain. If we assume a discrete solution and an equidistant domain discretization with N elements or nodal results, one can determine the discrete L2 norm of $\mathbf{b}$ compared to the exact solution $\hat{\mathbf{b}}$ as shown in equation (2) [11].

$$E_2 = \| \mathbf{b} - \hat{\mathbf{b}} \|_2 = \sqrt{\frac{1}{N} \sum_{n=1}^{N} \left(b_n - \hat{b}_n \right)^2} \tag{2}$$

3 THE PARTITIONED FSI FORMULATION

3.1 Framework

The present partitioned FSI framework in general consists of three programs: the computational structural mechanics (CSM) software $Carat++$ using total Lagrangian formulation, the computational fluid dynamics (CFD) program $OpenFOAM^{®}$ generally using an Euler formulation, and the coupling toolbox $Empire$. Using a Dirichlet-Neumann

partitioning, the surface displacements and traction forces are linked using mapping operations. The mentioned mapping operations on the one hand offer the possibility of non-matching grids at the common interface. On the other hand these operations produce additional (discretization) errors one has to care about.

3.2 Equations of State

The continuum equations of state are shown in equations (3-5). Herein, $\mathbf{d}$ in general represents displacements and $\mathbf{F}$ represents volume forces. ρ is the density and ν_f the kinematic viscosity. $\mathbf{\Phi}$ represents the fluid flux that in general is set to $\mathbf{\Phi} = \mathbf{u}$. The indices s and f indicate the structure resp. the fluid. Equation (3) represents the conservation of momentum of the structure with the first Piola-Kirchhoff stress tensor $\mathbf{P}$ [23]. Equations (4) represent the conservation of momentum and mass of the fluid with the absolute velocities $\mathbf{u}$ and the pressure p. Equations (5) show the compatibility of displacements $\mathbf{d}$ and the equilibrium of the traction forces $\mathbf{t}$ at the common interface surface Γ.

$$-\rho\frac{\partial^2\mathbf{d}}{\partial t^2} + \nabla\cdot\mathbf{P} + \rho\mathbf{F} = 0 \tag{3}$$

$$\frac{\partial\mathbf{u}}{\partial t} + \nabla\cdot(\mathbf{\Phi}\mathbf{u}) - \nabla\cdot(\nu_f\nabla(\mathbf{u})) + \frac{1}{\rho_f}\nabla p = 0 \qquad \nabla\cdot\mathbf{u} = 0 \tag{4}$$

$$\mathbf{d}_{s\Gamma} - \mathbf{d}_{f\Gamma} = 0 \qquad \mathbf{t}_{s\Gamma} + \mathbf{t}_{f\Gamma} = 0 \tag{5}$$

3.3 Discretization

The structural domain Ω is discretized with the Finite Element method (e.g. [1,22,23]). Equation (3) transforms to the weak form using the principle of virtual work with the virtual displacements $\delta\mathbf{d}$ given in equation (6) with the surface traction forces $\mathbf{T}$. The fluid domain V is discretized by the Finite Volume method (e.g. [5,9,10]) and equations (4) transform to the integral form given in equation (7) and (8). The interface equations (5) are discretized in the integral form using the Mortar method [16] with Galerkin weighted residuals with the test function $\tilde{\mathbf{d}}$ in equations (9).

$$\delta W = -\int_\Omega \rho\frac{\partial^2\mathbf{d}}{\partial t^2}\delta\mathbf{d}d\Omega + \int_\Omega \mathbf{S} : \delta\mathbf{E}d\Omega - \int_\Gamma \mathbf{T}\delta\mathbf{d}d\Gamma - \int_\Omega \rho\mathbf{F}\delta\mathbf{d}d\Omega = 0 \tag{6}$$

$$\int_V \frac{\partial\mathbf{u}}{\partial t}dV + \int_V \nabla\cdot(\mathbf{\Phi}\mathbf{u})\,dV - \int_V \nabla\cdot(\nu_f\nabla(\mathbf{u}))\,dV + \int_V \frac{1}{\rho_f}\nabla p dV = 0 \tag{7}$$

$$\int_V \nabla\cdot\nabla\frac{p}{\rho_f}dV - \int_V \nabla\cdot(\nabla\cdot(\mathbf{\Phi}\mathbf{u}))\,dV = 0 \tag{8}$$

$$\int_\Gamma \mathbf{d}_{s\Gamma}\tilde{\mathbf{d}}d\Gamma - \int_\Gamma \mathbf{d}_{f\Gamma}\tilde{\mathbf{d}}d\Gamma = 0 \qquad \int_\Gamma \mathbf{t}_{s\Gamma}\tilde{\mathbf{d}}d\Gamma + \int_\Gamma \mathbf{t}_{f\Gamma}\tilde{\mathbf{d}}d\Gamma = 0 \tag{9}$$

For moving boundaries of the fluid the Arbitrary-Eulerian-Lagrangian (ALE) formulation is used [21]. After application of Leibnitz' law and a few reformulations equations (7) and (8) still hold. The only change occurs in the flux $\mathbf{\Phi} = \mathbf{u} - \mathbf{u}_{grid}$ instead of $\mathbf{\Phi} = \mathbf{u}$.

3.4 Formal Orders of Accuracy

Following [24] for the structural dynamics in this contribution, using bilinear form functions for the membrane and a Newmark-Beta time integration method, the displacements converge with a formal order of p=2 in space and time. Following [8, 10] for fluid dynamics, the pressure and the velocities are discretized with formal second order schemes in space and time. Therefore the solution converges with a formal order of p=2 in a static domain. As soon as the boundaries of the domain are moving, the current software is "only" converging with a formal order of p=1, because the grid velocity is only approximated with a first order discretization. This small lack in accuracy lowers the complete formal order of the solution to p=1 as soon as the grid is moving.

The source/load sampling and all mapping procedures conducted during the process shown in figure 2 on the following page are introducing additional error. Therefore, all sampling and mapping processes have to be at least as accurate as the used discretization accuracy of the individual fields. The execution of these operations actually are only supporting the solution process of equations (6-9). Nevertheless, if there are errors with a lower order, or even worse mistakes, in these operations, one is not able to successfully assess the accuracy of the actual solution process. The currently used mapping and the sampling methods use bilinear form functions. Therefore they are similar to the used membrane element discretization, which is formally second-order accurate.

4 APPLICATION OF THE MMS

The framework for the application of the MMS on the given FSI procedure is shown in figure 2 on the next page. After choosing/inventing manufactured target solutions for all fields (step 1) one has to plug them into the individual differential equations in the differential form (equations (3)-(5)) to derive the non-zero solutions as source resp. force terms for the codes to be assessed using $Maple^{®}$. The boundary and initial conditions of all fields have to be calculated as well (step 2) from the target solution. The next step contains the implementation and sampling of the sources/forces in the codes (step 3). Next, the FSI problem is solved in its actual way (steps 4a and b). This test is repeated for all defined refinement stages in space and time (in general 5-7 refinements with a factor of r = 2). The resulting solutions of the fields are mapped in each simulation to a set of probes, which makes the results of all simulations comparable (step 5). Afterwards, an error evaluation using $Matlab^{®}$ is performed to calculate the errors and the observed orders of accuracy p (cf. equation (2) on page 4). The blue lined arrows in figure 2 on the next page denote a mapping process using Empire.

The modular arrangement shown in figure 2 on the following page gives the opportunity to perform a complete FSI investigation, but also individual CSM (skipping 4a) or CFD (skipping 4b) investigations optional with MMS, without changing the actual code. Additionally it is capable to assess the mapping (blue lined arrows in figure 2) and sampling routines (in steps 3 and 4 of figure 2) independent of the field solutions. To assess

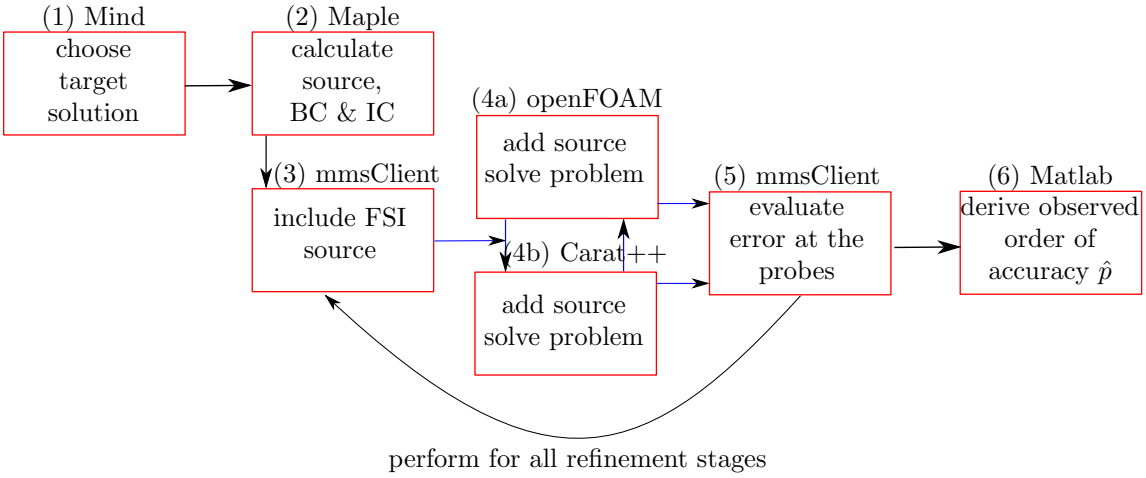


Figure 2: MMS Framework for the partitioned FSI environment

the sampling and the mapping of fields, one skips steps 4 and performs the arrangement of figure 2 for an analytical manufactured solution. This should be performed at the one hand for equal surface grids (matching grids) and at the other hand for non-matching grids. The successful assessment of these operations is an essential prerequisite for the testing of the FSI process.

Having a look at the equations of state (3-5) resp. (6-8), one has to decide where to put sources/forces for MMS. Often there are some limitations of the programs that some equations can't or shouldn't be accessed (e.g. [4]). In the present case, sources are added to the equations (6-8) and (9b).

5 MMS BENCHMARKS

Due to the length limitations of this contribution only three MMS benchmark examples can be shortly shown. Therefore the given set actually is only a subset of the named stairway of complexity to assess all parts of the code(s) individually, if possible.

5.1 Benchmark 1: CSM

The chosen example for CSM is an initially curved and pre-stressed membrane which will be deformed out-of-plane in an unsteady way (cf. table 1 on the following page and figure 3). This example assesses the fully geometric nonlinear formulation containing all geometric transformations, the mass/inertia contribution, the material law and the time integration. B is the thickness, E the young's modulus, ν_s the Poisson's ratio, S_{ps} the pre-stress, and t the time. A detailed discussion and more examples to CSM can be found in [7].

Table 1: Overview of benchmark example for CSM

init. config.	deformation	material		domain
$x = \theta^1$	$d_x = 0$	$E = 1000.0$	$B = 0.001$	$\theta^1 \in [0..1]$
$y = \theta^2$	$d_y = 0$	$\rho_s = 1000.0$	$S_{ps,\theta^1} = 25.0$	$\theta^2 \in [0..1]$
$z = \theta^1 - \theta^{1^2}$	$d_z = \frac{1}{4} sin\left(\theta^1 \pi\right) cos\left(\theta^2 \pi\right) sin\left(\frac{\pi}{2} t\right)$	$\nu_s = 0.30$	$S_{ps,\theta^2} = 25.0$	$t \in [0..2]$

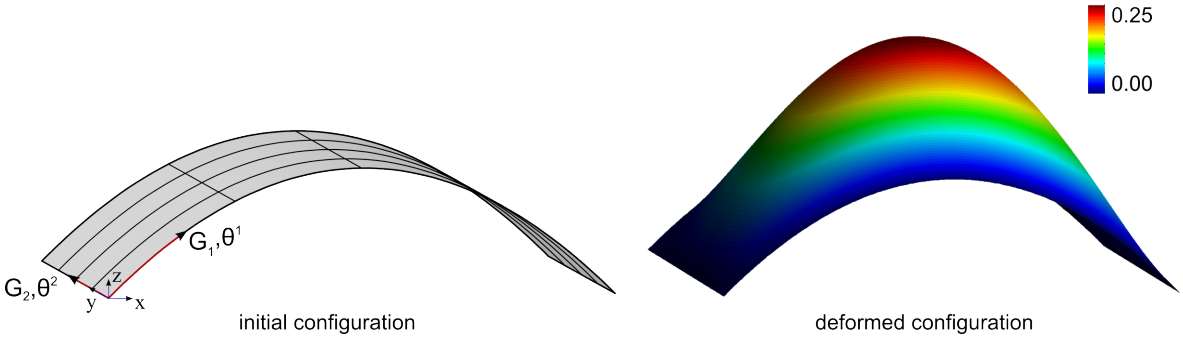


Figure 3: Initial and deformed configuration of the benchmark for CSM

5.2 Benchmark 2: CFD 2D with moving boundaries

The example for CFD is a laminar case with mesh motion of an initially rectangular domain. Due to the motion all cells get skewed and the cell faces get non-orthogonal. Therefore all geometric effects get exercised in the unsteady regime (cf. table 2). The only limitation is that the case is two dimensional. ν_f is the kinematic viscosity, u_g the grid velocity and ρ_f the density of the fluid. A detailed discussion and more examples to CFD can be found in [6].

Table 2: Overview of benchmark example for CFD with moving boundaries

fields	grid velocities and pressure	material	domain
$u = 1.0 + u_g$	$u_g = \pi cos\left(\frac{\pi}{2} y\right) cos\left(\pi t\right)$	$\nu_f = 0.50$	$x \in [0..1]$
$v = 2.0 + u_g$	$v_g = \frac{\pi}{2} \cdot sin\left(\pi x\right) cos\left(\frac{\pi}{2} y\right) cos\left(\pi t\right)$	$\rho_f = 1.0$	$y \in [0..1]$
	$p = 2 + 100 cos\left(x + y\right) sin\left(\pi t\right)$		$t \in [0..1]$

5.3 Benchmark 3: Sampling and Mapping Operations

The same operations are performed for (vector) field sampling, no matter if it's a force, a displacement, or a velocity field. Therefore, as described in section 4 one has to choose an analytical manufactured field solution for any field (cf. table 3). As the sampling and mapping are purely geometric operations a steady solution is sufficient.

Table 3: Overview of benchmark example for sampling and mapping

init. config.	field	domain
$x = \theta^1$	$f_x = 4 \cdot sin\left(\theta^1 \pi\right) \cdot cos\left(\theta^2 \pi\right)$	$\theta^1 \in [0..1]$
$y = \theta^2$	$f_y = sin\left(\theta^1 \pi\right) + cos\left(\theta^2 \pi\right)$	$\theta^2 \in [0..1]$
$z = \theta^1 - \theta^{1^2}$	$f_z = 16 \cdot sin^2\left(\theta^1 \pi\right) cos^2\left(\theta^2 \pi\right)$	

5.4 Benchmark 4: Fully Coupled FSI

The FSI benchmark consists of an initially flat and pre-stressed elastic membrane at the bottom of a 3D fluid cube. The out-of-plane deformation of the membrane exercises all geometric operations in the fluid and the structure (cf. table 4). The surface meshes of the fluid and the structure are once equally and once differently discretized. The absolute velocities u,v,w have exactly the grid velocity, and the pressure gradient is zero at the interface. This is guaranteed by the expressions in square brackets [15].

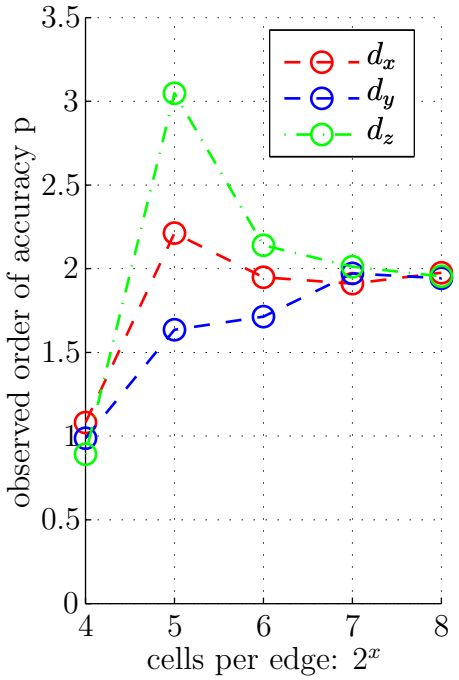
Table 4: Overview of benchmark example for FSI

domain	fields	material	
$x = \theta^1$	$d_x = 0 \quad d_z = 0$	$E = 1.0 \cdot 10^3$	$B = 0.001$
$y = 0$	$d_y = sin\left(\theta^1 \pi\right) sin\left(\pi t\right) \cdot t^2$	$\rho_s = 1.0 \cdot 10^6$	$S_{ps,\theta^1} = 25$
$z = \theta^2$	$u = w = 0.1 sin\left(x + y + z\right)\left[y - d_y\right] \quad v = \frac{\partial d_y}{\partial t}$	$\rho_f = 1.0 \cdot 10^3$	$S_{ps,\theta^2} = 25$
$\theta, \mathbf{x}, t \in [0..1]$	$p = -\frac{1}{2} + \left(cos\left(x + y\right) + sin(z)\right) sin\left(\pi t\right)\left[y - d_y\right]^2$	$\nu_f = 0.5$	$\nu_s = 0.30$

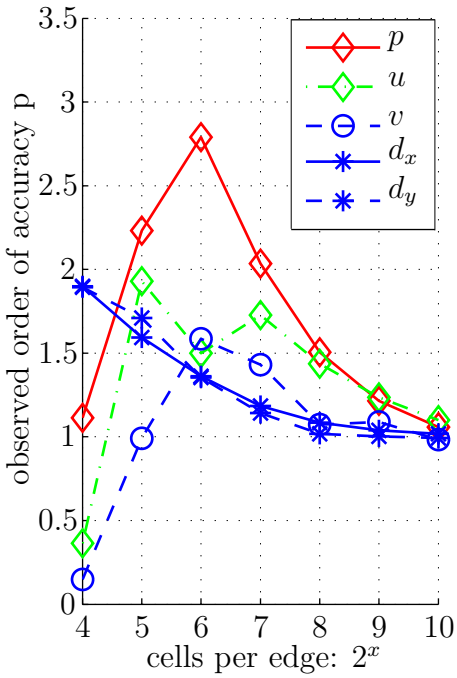
6 RESULTS & CONCLUSIONS

The resulting observed order of accuracy of the given benchmarks using the shown environment are shown in figure 4. One observes almost the individual formal orders of accuracy (cf. 3.4). As already mentioned, the accuracy of the fluid solution (and his forces) drop the total formal and the observed order of accuracy to 1 (figure 4d).

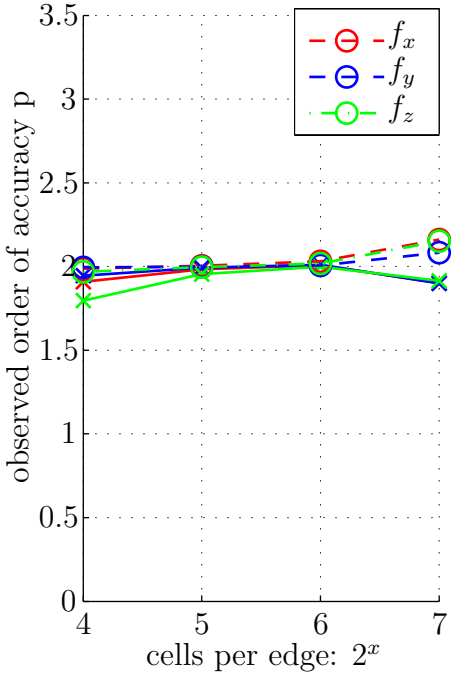
In this paper, a benchmark test suite for a partitioned FSI environment was developed and successfully applied. Especially the topic of non-matching surface grids was addressed. The results give confidence that all assessed software components each on its own, but also all together work as intended. Therefore the environment at the one hand can be used for further investigations for physically realistic simulations mentioned in the introduction. At the other hand the environment could be improved at the 'weak points', measured at the formal orders of accuracy. This means, that improvements should take place, if possible, at the fluid software, as it is the only chain link which is 'only' first order accurate in the process. To finally produce high-quality results with the complete FSI process, the environment should produce second order accurate results. Beyond the pure meaning of providing benchmarks, this contribution can be considered as a general and



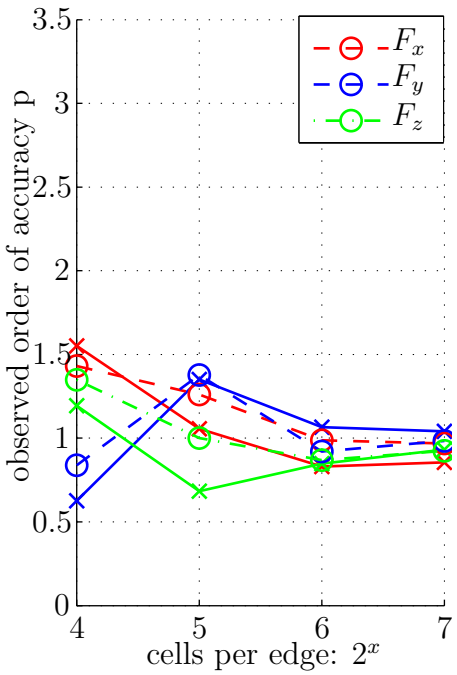
(a) benchmark 1: displacement convergence



(b) benchmark 2: fluid convergence



(c) benchmark 3 displacement convergence



(d) benchmark 4: force convergence

Figure 4: Benchmark results using the presented environment of section 4

generic frame for the reader to transfer or to extend the cases to his software environment and finally to generate own customized or tailored benchmarks. As the confidence in the environment of the numerical wind tunnel is established, one will be able to discuss specific issues that surpass the limitations of the experimental wind tunnel. As an example for these investigations one could think of: the capability to identify sensitivities and crucial parameters of the interaction between the wind and lightweight structures; the assessment of numerical and physical modeling aspects; uncertainty quantification.

REFERENCES

[1] K.-J. Bathe. *Finite element procedures*, volume 2. Prentice-Hall Englewood Cliffs, 1996.

[2] F. G. Blottner. Accurate navier-stokes results for the hypersonic flow over a spherical nosetip. *Journal of spacecraft and Rockets*, 27(2):113–122, 1990.

[3] L. Eca, M. Hoekstra, A. Hay, and D. Pelletier. On the construction of manufactured solutions for one and two-equation eddy-viscosity models. *International journal for numerical methods in fluids*, 54(2):119–154, 2007.

[4] S. Étienne, A. Garon, and D. Pelletier. Some manufactured solutions for verification of fluid-structure interaction codes. *Computers & Structures*, 106-107:56–67, 2012.

[5] J. H. Ferziger and M. Perić. *Computational methods for fluid dynamics*, volume 3. Springer Berlin, 1996.

[6] R. Fisch, J. Franke, R. Wüchner, and K.-U. Bletzinger. Code verification of openfoam® solvers using the method of manufactured solutions. In *7th OpenFOAM Workshop Darmstadt*, 2012.

[7] R. Fisch, J. Franke, R. Wüchner, and K.-U. Bletzinger. Code verification examples of a fully geometrical nonlinear membrane element using the method of manufactured solutions. *Proceedings Structural Membranes 2013*, 2013.

[8] R. Issa. Solution of the implicitly discretised fluid flow equations by operator-splitting. *Journal of Computational physics*, 62(1):40–65, 1986.

[9] H. Jasak. *Error analysis and estimation for the finite volume method with applications to fluid flows*. PhD thesis, Imperial College London, 1996.

[10] F. Juretic. *Error analysis in finite volume CFD*. PhD thesis, Imperial College London (University of London), 2005.

[11] P. Knupp and K. Salari. *Verification of computer codes in computational science and engineering*. CRC Press, 2003.

[12] M. Kojić and K.-J. Bathe. Studies of finite element procedures—stress solution of a closed elastic strain path with stretching and shearing using the updated lagrangian jaumann formulation. *Computers & Structures*, 26(1):175–179, 1987.

[13] W. Oberkampf and T. Trucano. Validation methodology in computational fluid dynamics. *Sandia National Labs., Albuquerque, NM (US); Sandia National Labs., Livermore, CA (US)*, 2000.

[14] W. Oberkampf and T. Trucano. Verification and validation in computational fluid dynamics. *Progress in Aerospace Sciences*, 38(3):209–272, 2002.

[15] W. L. Oberkampf and C. J. Roy. *Verification and Validation in Scientific Computing*. Cambridge University Press, 2010.

[16] M. A. Puso and T. A. Laursen. A mortar segment-to-segment contact method for large deformation solid mechanics. *Computer Methods in Applied Mechanics and Engineering*, 193(6):601–629, 2004.

[17] P. Roache. *Verification and validation in computational science and engineering*. Hermosa Publishers, 1998.

[18] C. Roy, C. Nelson, T. Smith, and C. Ober. *Verification of Euler/Navier–Stokes codes using the method of manufactured solutions*, volume 44. Wiley Online Library, 2004.

[19] K. Salari and P. Knupp. Code verification by the method of manufactured solutions. *Sandia National Labs., Albuquerque, NM (US); Sandia National Labs., Livermore, CA (US)*, 2000.

[20] S. Schlesinger, R. Crosbie, R. Gagné, G. Innis, C. Lalwani, J. Loch, R. Sylvester, R. Wright, N. Kheir, and D. Bartos. Terminology for model credibility. *Simulation*, 32(3):103–104, 1979.

[21] P. Schreurs, F. Veldpaus, and W. Brekelmans. Simulation of forming processes, using the arbitrary eulerian-lagrangian formulation. *Computer methods in applied mechanics and engineering*, 58(1):19–36, 1986.

[22] G. Strang and G. J. Fix. *An analysis of the finite element method*, volume 212. Prentice-Hall Englewood Cliffs, NJ, 1973.

[23] P. Wriggers. *Nonlinear finite element methods*. Springer, 2008.

[24] O. C. Zienkiewicz and R. L. Taylor. *The finite element method*, volume 3. McGraw-Hill London, 1977.