V European Conference on Computational Fluid Dynamics
ECCOMAS CFD 2010
J. C. F. Pereira and A. Sequeira (Eds)
Lisbon, Portugal,14-17 June 2010

PARTITIONED FLUID-STRUCTURE INTERACTION SIMULATIONS USING A HIERARCHICAL CARTESIAN FLOW SOLVER

Miriam Mehl[†,*], Bernhard Gatzhammer[†] and Tobias Neckel[†]

[†]Department of Computer Science, Technische Universität München
Boltzmannstraaße 3, 85748 Garching, Germany
e-mail: {mehl,gatzhamm,neckel}@in.tum.de

Key words: Fluid-Structure Interaction, Partitioned Approach, Hierarchical Cartesian Grids, Multilevel

Abstract. *Partitioned approaches for the simulation of multi-physics scenarios allow for a fast solver implementation but still pose high computational challenges. In addition, sophisticated numerical methods are required to achieve stable simulations for time-dependent applications. This in particular holds for fluid-structure interactions with incompressible fluids. As the bottleneck is the flow solver, here, we propose a highly memory efficient flow solver based on our Cartesian PDE framework Peano. The underlying tree-structured fixed grids allow for the efficient handling of large geometry or even topology changes and for a hierarchical multilevel coupling with the structure solver. The inhouse coupling tool preCICE takes over the whole coupling control and makes the involved solver exchangable independent from each other and without loosing the implemented coupling method.*

1 INTRODUCTION

Partitioned approaches allow for a fast implementation of an environment for fluid-structure interaction simulations using established solvers for fluid and structure scenarios. The price for this simplicity is the need for sophisticated coupling strategies to achieve a stable simulation. In addition, the partitioned approach still poses high computational challenges on the used codes due to for examples complex and moving geometries, small time steps, high grid resolutions, repeated coupling iterations, etc. Among the two solvers involved in fluid-structure simulations, the flow solver in general is the bottleneck consuming the majority of the runtime of the whole coupled simulation.

The structuredness of adaptive Cartesian grids allows to develop highly memory-efficient solvers for various types of PDEs. Our flow solver Peano based on such grids will be proposed in a separate contribution in these proceedings. First results achieved with an older version of the solver have also been presented in [5]. The current presentation focusses on the potential of the Peano solver in the context of partitioned fluid-structure

interactions using our inhouse tool preCICE (formerly called FSI❄ce) [9]. We will shortly show the highly efficient realisation of large geometry or even topology changes in an Eulerian (fixed grid) setting. An additional big advantage of our implementation of adaptive Cartesian grid algorithms in the flow solver Peano is the inherent hierarchy of grid levels, each of which are accessible separately in an efficient way. Thus, algorithms taking into account different coupling requirements on different grid levels corresponding to different characteristic Fourier modes become feasible. Section 2 describes the properties of the Peano flow solver that are important for partitioned fluid-structure simulations. The coupling tool preCICE and the main concepts behind are presented in Sect. 3. Section 4 sketches a concept for multilevel coupling methods and shows the potential of our combination Peano plus preCICE in such a setting.

2 THE FLOW SOLVER IN PEANO

As mentioned above, we intend to overcome the very high computational and memory requirements of the flow solver as well as stability problems of the partitioned simulation. In this contribution, we focus on the aspects of our flow solver implemented in the inhouse PDE framework Peano that are helpfull with respect to these tasks. A more detailed description of the framework, the flow solver and the ideas behind can be found in [11, 6, 26, 18].

Our PDE framework Peano works on tree-structured adaptive Cartesian grids (see Fig. 1 for an example). The structuredness of these grids minimizes memory requirements as it makes the storage of all kinds of relations between grid elements such as 'vertex belongs to edge', 'edge belongs to face', face belongs to element' etc. obsolete. Combining these grids with a cell-wise operator evaluation even eliminates the need to store neighbor relations of grid elements and specialized operator stencils at boundaries between different refinement levels: all operators are evaluated cell-wise using only data available in a cell. The operator-contributions of all four (2D) or eight (3D) neighboring cells accumulate to the final operator value at a grid vertex, e.g. In case of our flow solver, cell data are velocities at the grid vertices and pressure at the cells midpoints. Hanging nodes do not pocess degrees of freedom but, instead, the values at hanging nodes are interpolated from neighboring non-hanging nodes of the respective father cell. For more details on this cell-wise operator evaluation see [3, 11]. The resulting memory requirements of the Peano flow solver are 14 bytes per cell and 20 bytes per vertex in 2D or 18 bytes per cell and 28 bytes per cell in 3D, respectively. As Peano in addition uses a sophisticated data handling using solely streams and stacks based on a space-filling curve traversal of the grid, also the cache-efficiency has been optimised. Hitrates in level 2 cache and above are typically about 99% or higher [11, 26].

Moving and deforming geometries occuring in fluid-structure interaction as a result of the forces exerted on the structure by the fluid require sophisticated methods for grid adaption in both the fluid and the structure solver. Whereas the structure solver in general only has to handle moderate geometry changes, the flow solver faces much larger changes

up to even topology changes. To illustrate this, just think of the example of a small particle advected in a fluid flow through a channel: the particle shape does not change at all whereas the movement of the particle – maybe even touching the channel wall and rotating – strongly changes the shape of the fluid domain. Due to this fact, Lagrangian moving grid approaches are convenient for structure solvers. For flow solvers, many approaches have been developed to combine a good tracing of the actual geometry with a good grid quality and, of course, low computational costs for grid changes. Arbitrary Langrangian Eulerian approaches [12, 13, 20, 21] move boundary grid points at the structure surface according to the structure movement and deformation and adapt the grid in the inner domain via interpolation or solution of a partial differential equation ensuring suitable numerical properties of the resulting grid. However, for large geometry changes, these approaches require frequent remeshing of the domain as mesh deformation becomes too large. In the last years, however, a trend towards Eulerian fixed grid methods in the field of fluid-structure interaction simulations has been observed [2, 15, 25]. For these methods, grid elements and nodes do not change. Instead, the information about the actual geometry is stored in the grid elements. Various methods such as immersed boundary methods [17], cut-cell approaches [8], level-set methods [19], or extended finite elements [10] allow to achieve a good accuracy in the geometry approximation for these fixed grids although grid lines or faces are of course in general not aligned with domain boundaries.

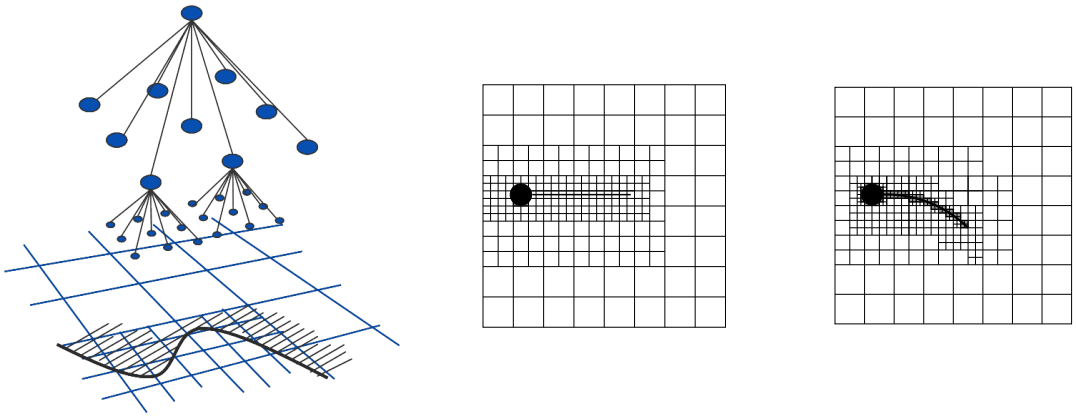

Figure 1: Left: simple two-dimensional example of a tree-structured adaptive Cartesian grid with a partitioning into three parts per refinement step and coordinate direction. Middle and right: Eulerian grid for the fluid domain in a fluid-structure interaction problem. Grid nodes don't move but adaptive refinements 'follow' the changing geometry.

In Peano, the tree-structured grids adapt very efficiently to geometry changes: Peano works with a marker-and-cell approach and, thus, only has to change cell-markers (fluid or obstacle) and the local grid refinement according to a geometry change. Figure 1 shows (very coarse) computational grids for two different times of a fluid-structure interaction simulation. As geometry changes are local, in general, only a subtree of the whole tree defining the grid has to be changed in most cases. Thus, one grid traversal is the upper

bound for the costs of the fluid grid change as a response of a structure movement.

3 THE COUPLING TOOL PRECICE

Our inhouse coupling tool preCICE [9] was developped (under the previous name FSI❄ce [4]) in order to provide a flexible and easiliy expandable tool for partitioned fluid-structure interaction simulations.

The difficulties for such tools are twofold: On the one hand, the theoretical flexibility of the partitioned approach, that is the possibility to use arbitrary fluid and structure solvers and to exchange them easily if required, should be maintained. On the other hand, sophisticated coupling strategies and data mapping methods for non-matching grids have to be provided in order to achieve a stable and accurate simulation.

preCICE is designed as an own numerical unit that centralizes the complete coupling functionality including both data mapping and coupling control (see Fig. 2). To hide solver details from the partner solver, all data that are to be exchanged at the fluid-structure interface are mapped onto a central coupling surface triangulation. Such, a solver once being linked to that interface can be coupled with any other solver without further (technical) efforts.

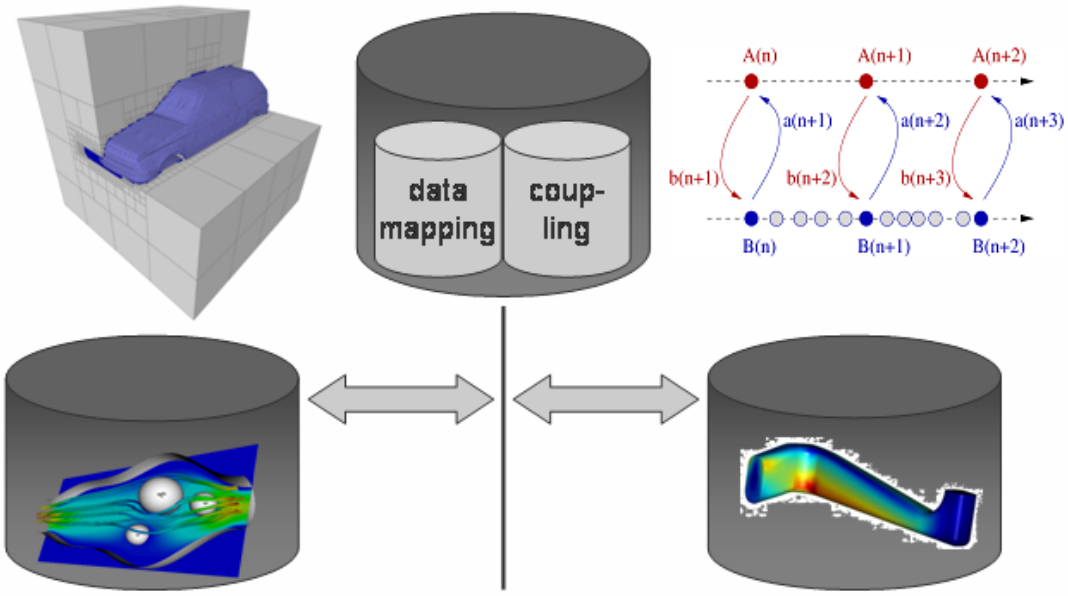

Figure 2: Concept of our coupling tool preCICE offering a centralized data mapping and coupling control functionality.

The most costly part of the data mapping is the determination of neighborhood relations or element intersections of the boundary description of a solver and the central surface triangulation. preCICE provides a highly efficient support for these tasks based

on quad- (in 2D) or octrees (in 3D) as a fast data access mechanism [4, 9]. Currently, rather simple projection and linear interpolation methods are implemented. However, further data mappings can be added via clearly defined interfaces.

The coupling control, that is the realization of a couping strategy is a central component of preCICE. Unlike in other approaches using MpCCI [14], for example, preCICE provides the complete functionality ranging from the definition of the iteration scheme, over numerical work to be done at the fluid-structure interface, to convergence control. Thus, this functionality can be used regardless of the actually chosen solvers. Again, currently implemented versions are limited to simple methods, that is explicit or implicit staggered coupling with subcycling in the solvers if required. The potential of preCICE, however, lies in the possibility to realize far more complex coupling strategies such as the multilevel approaches described in Sect. 4.

Technically, preCICE is realized as a library with a clearly defined application programming interface. Data to be communicated between the solvers are coupling control parameters such as time step and convergence information as well as data at the fluid-sructure interface. The latter is first mapped to the central interface triangulation and then send to the partner solver. preCICE implements various communication possibilities using MPI [16]: The first possibility is the definition of separate global communication spaces within the solvers. A second possibility works similar to socket-based communication and uses MPI 2.0 features to start the solvers independently and to setup the communication via preCICE. This possibility has the drawback that MPI 2.0 is typically not yet available on supercomputers. The last possibility is to compile the solver codes into one executable and to start this with several processes.

4 MULTILEVEL COUPLING CONCEPTS

It is known that multilevel coupling approaches can substantially speed up the convergence of partitioned fluid-structure simulations [1, 27]. For incompressible fluids, we even observe different stability restrictions of a staggered iteration for different fourier modes of the structure deformation:

[22] examines a simple two-dimensional example with a one-dimensional structure displayed in Fig. 3. A fluid domain of length L in x-direction, a flexible beam at the lower bound and an infinite extension in upper direction is considered. It is assumed that a subiteration (Gauss-Seidel) method solves the continuous fluid and structure equations exactly over a time interval δt and exchanges data after the computation of this interval:

$$z_{n+1} \quad = \quad \text{structure\_solve}(\text{fluid\_solve}(z_n)).$$

Here, structure_solve and fluid_solve denote the computation of the exact solution of the structure or fluid equation over the time interval using the pressure or displacement at the beam as input parameter. z_n is the didplacement of the beam after iteration n.

This yields the contraction number

$$\rho(k,\omega) = k \cdot \frac{\pi^{-1}\rho_0 L}{m} + O(\omega^{-1}),$$

for an isolated error mode

$$e_n(x,t) = \hat{e}(k,\omega)\exp\left(ik\frac{\pi}{L}x + i\omega t\right),$$

where k denotes the spatial error frequency relative to the beam length, ω the time frequency of the beam deformation, ρ_0 the fluid density, and m the mass of the beam per unit length. The conclusion drawn from these results in [22] is that the subiteration process becomes unstable independent from the time step size δt for big values of $\frac{\rho_0}{m}$. However, if we consider $\rho(k,\omega)$ not only for the 'worst' frequency $k = 1$, we see in addition that the subiteration process acts as a smoother for the coupled iteration: High frequency errors are reduced fastly by a low contraction number, whereas low frequencies are difficult. However, the analysis does not give any hints on the dependence of the convergence on the spatial resolution of the used solver grids yet.

[7, 24] examine a fluid flow through an axisymmetric flexible tube (see Fig. 3) and also perform a one-dimensional Fourier analysis of a partitioned subiteration solver, but they take into account the time and spatial discretization of the solvers. The analysis shows that the amplification factor of the error modes increases with

1) decreasing spatial frequency of the mode,

2) decreasing time step size, and

3) decreasing spatial grid resolution.

This means in particular that, in contrast to standard multigrid theory for sparse linear systems, low frequencies do not become easier to handle on spatially coarses grids. The difficulties even become worse with decreasing spatial resolution. [24] and citations therein show that the convergence of the partitioned iterations can be improved by a quasi-Newton method. They suggest to compute the Jacobian only for the unstable (low) frequencies. This makes the following multilevel solver for the partitioned problem promising:

- perform (quasi-)Newton iterations on coarse grids such that all modes visible at this spatial resolution are damped well,
- interpolate to finer grids,
- and include those new modes on the fine grid in the Jacobian that are still unstable, solve others with Gauss-Seidel iterations,
- refine further...

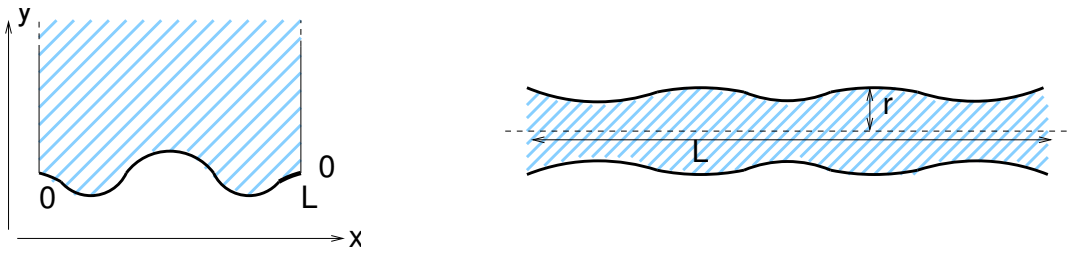

Figure 3: Schematic view of two example problems: left: example examined in [22]. The fluid domain of length L in x-direction is bounded at the lower bound by a flexible beam and unbounded in upper direction. Right: example considered in [7, 24]. An axisymmetric flexible tube of length L and local radius r.

With such an approach, we could profit from the much cheaper iterations on coarse grids to reduce the problematic frequencies.

With a combination of our coupling tool preCICE and the flow solver in Peano, we can realize this multilevel method in an ideal way: As a separate numerical entity, preCICE can implement a multilevel version of interface displacement, which in particular requires hierarchization and dehierarchization methods (some might call this restriction and interpolation). In addition, preCICE could compute the Jacobians (or their inverses) using for example a reduced order approach such as proposed in [23]. Our flow solver in Peano is capable of performing simulations on a selected grid level (without loosing memory efficiency) with a multilevel full approximation scheme [26]. Figure 4 illustrates the multilevel interaction between preCICE and the Peano flow solver in a schematic way.

5 SUMMARY

We have proposed our realization of two out of three necessary components for a partitioned fluid-structure interaction simulation: The flow solver in the PDE framework Peano and the inhouse coupling tool preCICE. Both have been designed in order to achieve maximal numerical and hardware efficiency and, at the same time, maintain the theoretical flexibility of the partitioned approach to be able to combine arbitrary solvers with each other and with various coepling strategies and to easily exchange one or several of these ingredients if required. Based on the analysis of coupling properties for different Fourier modes of structure deformation found in literature, we have sketched ways to efficient multilevel coupling approaches and shown the potential of preCICE and Peano in this context. First results for these concepts will be shown at the conference.

REFERENCES

[1] H. Bijl, A. H. van Zuijlen, and S. Bosscher. Two level algorithms for partitioned fluid-structure interaction computations. In P. Wesseling, E. Oñate, and J. Périaux, editors, *ECCOMAS CFD 2006, European Conference on Computational Fluid Dynamics*. TU Delft, 2006.

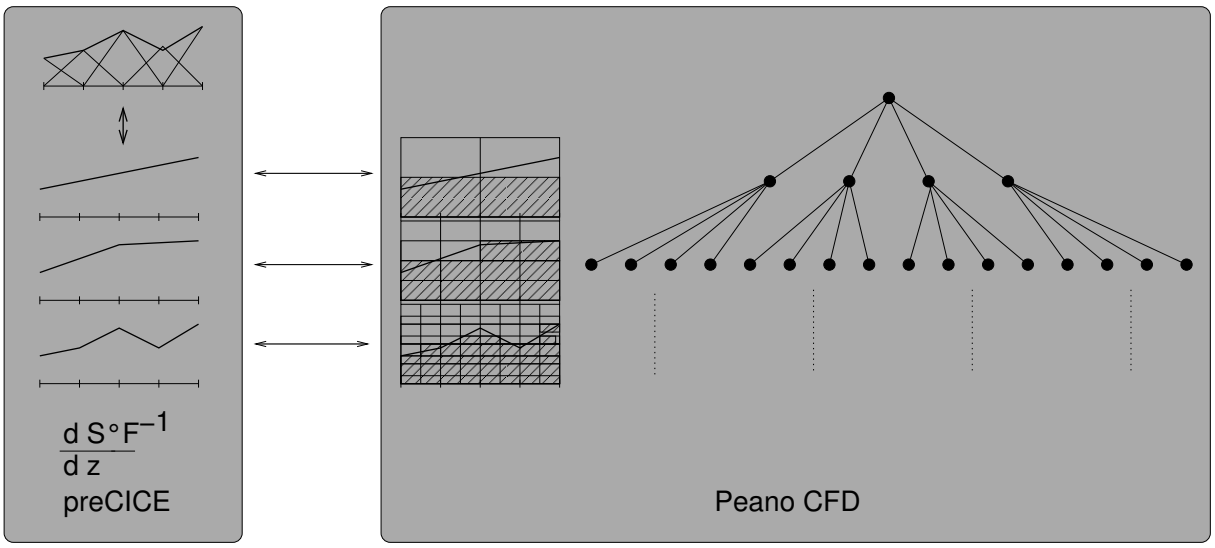

Figure 4: Schematic view of a multilevel partitioned solver realization using the flow solver in Peano and the coupling tool preCICE. The aibility of Peano to work on selected grid levels and the possibility to implement a interface hierarchization and dehierarchization in preCICE allow for an efficient implementation of the whole coupled environment. The structure solver side is neglected in this illustration.

[2] M. Bozkurttas, H. Dong., V. Seshadri, R. Mittal, and F. Najjar. Towards numerical simulation of flapping foils on fixed cartesian grids. In *43rd Aerospace Sciences Meeting and Exhibit*, 2005. AIAA 2005-0079.

[3] D. Braess. *Finite Elements—Theory, Fast Solvers, and Applications in Solid Mechanics*. Cambridge University Press, 3rd edition, 2007.

[4] M. Brenk, H.-J. Bungartz, M. Mehl, and T. Neckel. Fluid-structure interaction on cartesian grids: Flow simulation and coupling environment. In H.-J. Bungartz and M. Schäfer, editors, *Fluid-Structure Interaction*, number 53 in Lecture Notes in Computational Science and Engineering, pages 233–269. Springer, 2006.

[5] M. Brenk, H.-J. Bungartz, and T. Neckel. Cartesian discretisations for fluid-structure interaction – consistent forces. In P. Wesseling, E. Oñate, and J. Périaux, editors, *ECCOMAS CFD 2006, European Conference on Computational Fluid Dynamics*. TU Delft, 2006.

[6] H.-J. Bungartz, M. Mehl, T. Neckel, and T. Weinzierl. The pde framework peano applied to computational fluid dynamics. *Computational Mechanics*, 2009. accepted.

[7] J. Degroote, K.J. Bathe, and J. Vierendeels. Performance of a new partitioned procedure versus a monolithic procedure in fluid-structure interaction. *Comput. Struct.*, 87:793–801, 2009.

[8] F. Gao, D. M. Ingram, D. M. Causon, and C. G. Mingham. The development of a Cartesian cut cell method for incompressible viscous flows. Int. J. Numer. Meth. Fluids [online]. in press.

[9] B. Gatzhammer and M. Mehl. FSI❄ce – a modular simulation environment for fluid-structure interactions. In A. Meister, M. Schäfer, and S. Turek, editors, *Fluid-Structure Interaction: Theory, Numerics, and Applications*, Lecture Notes in Computational Science and Engineering. Springer-Verlag, 2009. accepted.

[10] A. Gerstenbrger and W. A. Wall. An extended finite element method / mortar method based approach for fluid-structure interactions. *Computer Methods in Applied Mechanics and Engineering*, 197:1699–1714, 2007.

[11] F. Günther, M. Mehl, M. Pögl, and C. Zenger. A cache-aware algorithm for PDEs on hierarchical data structures based on space-filling curves. *SIAM Journal on Scientific Computing*, 28(5):1634–1650, 2006.

[12] C. W. Hirt, A. A. Amsden, and J. L. Cook. An arbitrary lagrangian-eulerian computing method for all flow speeds. *J. Comp. Phys.*, 14:227–253, 1974.

[13] T. J. R. Hughes, W. K. Liu, and T. K. Zimmermann. Lagrange-eulerian finite element formulation for incompressible viscous fows. *Comput. Methods Appl. Mech. Engrg.*, 29:329–349, 1981.

[14] W. Joppich, M. Kürschner, and the MpCCI team. Mpcci — a tool for the simulation of coupled applications. *Concurrency Computat.: Pract. Exper.*, 18(2):183–192, 2006.

[15] M. Mehl. An eulerian divergence preserving approach for partitioned fluid-structure simulations on cartesian grids. In P. Wesseling, E. Oñate, and J. Périaux, editors, *ECCOMAS COUPLED PROBLEMS 2007, Proc. of the Thematic Conf. on Computational Methods for Coupled Problems in Science and Engineering*, pages 96–99. International Center for Numerical Methods in Engineering (CIMNE), 2007.

[16] MPI: A message-passing interface standard, version 1.1, 1995. Manual.

[17] R. Mittal and G. Iaccarino. Immersed boundary methods. *Annu. Rev. Fluid Mech.*, 37:239–261, 2005.

[18] T. Neckel. *The PDE Framework Peano: An Environment for Efficient Flow Simulations*. Verlag Dr. Hut, 2009.

[19] S. Osher and R. Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces*. Springer, 2003.

[20] J. Sarrate, A. Huerta, and J. Donea. Arbitrary lagrangian-eulerian formulation for fluid-rigid body interaction. *Comput. Methods Appl. Mech. Engrg.*, 190:3171–3188, 2001.

[21] M. Schäfer, M. Heck, and S. Yigit. An implicit partitioned method for numerical simulation of fluid-structure interaction. In H.-J. Bungartz and M. Schäfer, editors, *Fluid-Structure Interaction*, number 53 in Lecture Notes in Computational Science and Engineering, pages 171–194. Springer-Verlag, 2006.

[22] E.H. van Brummelen. Added mass effects of compressible and incompressible flows in fluid-structure interaction. (DACS-09-001), 2009.

[23] J. Vierendeels. Implicit coupling of partitioned fluid-structure interaction solvers using reduced-order models. In H.-J. Bungartz and M. Schäfer, editors, *Fluid-Structure Interaction*, number 53 in Lecture Notes in Computational Science and Engineering, pages 1–18. Springer-Verlag, 2006.

[24] J. Vierendeels. Stability issues in partitioned fsi calculations. In H.-J. Bungartz and M. Schäfer, editors, *Fluid-Structure Interaction – Part II*, Lecture Notes in Computational Science and Engineering. Springer-Verlag, 2010. to appear.

[25] P. Gamnitzer W.A. Wall and A. Gerstenberger. Fluid-structure interaction approaches on fixed grids based on two different domain decomposition ideas. *International Journal of Computational Fluid Dynamics*, 22(6):411–427, 2008.

[26] T. Weinzierl. *A Framework for Parallel PDE Solvers on Multiscale Adaptive Cartesian Grids*. Verlag Dr. Hut, 2009.

[27] S. Yigit, M. Heck, D. C. Sternel, and M. Schäfer. Efficiency of fluid-structure interaction simulations with adaptive underrelaxation and multigrid acceleration. *Int. J. Multiphysics*, 1:85–99, 2007.