

Coupling techniques for partitioned fluid-structure interaction simulations with black-box solvers

Joris Degroote¹, Robby Haelterman², Sebastiaan Annerel¹, Jan Vierendeels¹

¹ Ghent University, Department of Flow, Heat and Combustion Mechanics
Sint-Pietersnieuwstraat 41, B-9000 Ghent, Belgium

² Royal Military Academy, Department of Mathematics (MWMW)
Renaissancelaan 30, B-1000 Brussels, Belgium

Abstract

In partitioned simulations of fluid-structure interaction, the flow and the displacement of the structure are calculated separately and coupling iterations between the flow solver and the structural solver are required to calculate the solution of the coupled problem if the interaction is strong. This work is a comparison of three coupling algorithms which use the flow solver and structural solver as a “black box”. Consequently, these algorithms are suitable for implementation in future versions of MpCCI. It is demonstrated that the algorithm of the interface quasi-Newton technique with an approximation for the inverse of the Jacobian from a least-squares model is straightforward and that this technique needs a relatively low number of coupling iterations in the simulation of an oscillating flexible beam and the propagation of a pressure wave in a flexible tube.

Introduction

Recent research on fluid-structure interaction (FSI) has yielded both more complex applications and algorithmic improvements. Interesting applications can be found in aeronautics [1] and turbomachinery [2], where FSI is used to prevent flutter. Another challenging problem is dynamic analysis of parachutes [3] due to the thin, highly flexible structure. Also of great interest are biomedical applications like blood flow in arteries [4] and in artificial heart valves [5].

In partitioned simulations of fluid-structure interaction, the flow and the displacement of the structure are calculated separately. An existing flow solver is coupled with an existing structural solver and hence software modularity is preserved. Conversely, the monolithic approach requires a code developed for this particular combination of physical problems. Moreover, in the partitioned approach, the flow equations and the structural equations can be solved with different, possibly more efficient techniques which have been developed specifically for either flow equations or structural equations. On the other hand, a coupling algorithm is indispensable in partitioned simulations to take into account the interaction between the fluid flow and the motion of the structure.

If the interaction between the flow and the displacement of the structure is relatively weak, like in aeroelastic simulations, no coupling iterations are necessary and the flow and the displacement of the structure have to be calculated only once or twice within every time step to find the coupled solution

[1][6][7][8]. In cases with strong interaction, like in simulations of arteries and other applications with incompressible fluids, several coupling iterations have to be performed within a time step and the convergence of these coupling iterations requires an appropriate coupling algorithm.

The coupled problem can be solved in a partitioned way with fixed-point iterations using Aitken relaxation [9], with Newton iterations and with several other techniques. However, obtaining the exact Jacobian that is required for Newton iterations is difficult or even impossible if the solvers are “black box” commercial codes. Fortunately, several quasi-Newton techniques exist to approximate the Jacobian, e.g. interface quasi-Newton iterations with an approximation for the inverse of the Jacobian from a least-squares model (IQN-ILS) [10] and interface block quasi-Newton iterations with least-squares approximations for the Jacobian of the flow solver and structural solver (IBQN-LS) [11]. Black-box solvers can thus be coupled in several ways which is advantageous for both the users and the providers of commercial codes.

In this work, the implementation and the performance of the IQN-ILS and IBQN-LS quasi-Newton algorithms is compared to Aitken relaxation. Two cases are analyzed, that is the oscillation of a flexible beam attached to a rigid cylinder in a laminar flow [12] and the propagation of a pressure wave in a three-dimensional tube which is a simplified model for an artery [13]. The results demonstrate that both the IQN-ILS and IBQN-LS technique are significantly faster than Aitken relaxation and the comparison of the algorithms shows that the IQN-ILS algorithm can easily be implemented. All simulations have been performed with Fluent (Ansys, Inc.) and Abaqus (Simulia, Inc.) which have been coupled by means of a coupling code. It would be a step forward for the fluid-structure interaction community if it would be possible to implement these algorithms in future versions of MpCCI.

Definitions

This section gives an abstract definition of the flow solver and structural solver to emphasize that the solvers are treated as black boxes. The function

$$\mathbf{y} = \mathbf{F}(\mathbf{x}) \quad (1)$$

is referred to as the flow solver and it concisely represents several operations. The discretized position $\mathbf{x} \in \mathbb{R}^u$ of the fluid-structure interface is given to the flow code, and the grid of the fluid domain adjacent to the interface is adapted accordingly. Subsequently, the flow equations are solved for the fluid state in the entire fluid domain, which also results in a stress distribution $\mathbf{y} \in \mathbb{R}^w$ on the interface. The structural solver is represented by the function

$$\mathbf{x} = \mathbf{S}(\mathbf{y}) \quad (2)$$

This expression indicates that the fluid stress distribution on the interface is given to the structural code which then calculates the position of the entire structure and thus also the new position of the fluid-structure interface. With these definitions, the FSI problem is given by

$$\mathbf{x} = \mathbf{S} \circ \mathbf{F}(\mathbf{x}) \quad \text{or} \quad \mathbf{R}(\mathbf{x}) = \mathbf{S} \circ \mathbf{F}(\mathbf{x}) - \mathbf{x} = \mathbf{0} \quad (3)$$

in fixed-point or root-finding formulation, respectively, with R being the residual operator.

Generally, the position of the fluid-structure interface and the stress on the interface will not be discretized in the same way by the flow solver and by the structural solver and hence a mapping or interpolation has to be performed when data is exchanged between the solvers. However, coupling algorithms and data transfer algorithms are separate research topics and only coupling algorithms are considered in this work. Hence, matching meshes are used for the fluid-structure interface. This does not limit the applicability of the coupling algorithms as one can always include a mapping or interpolation inside the functions F and/or S .

In the remainder of this paper, all values and functions are at the new time level $(n + 1)$, unless indicated otherwise with a left superscript. A right superscript indicates the coupling iteration while a subscript denotes the element in a vector. Capital letters denote matrices; bold lower case letters indicate vectors and lower case letters represent scalars. Approximations are indicated with a hat. The output of the solvers F and S is indicated with a tilde because this is only an intermediate value that is not passed on to the next coupling iteration. This tilde is dropped once the final value that will be used in the next iteration has been calculated. All coupling algorithms begin a time step from an extrapolation of the interface's position

$$\mathbf{x}^0 = \frac{5}{2}({}^n\mathbf{x}) - 2({}^{n-1}\mathbf{x}) + \frac{1}{2}({}^{n-2}\mathbf{x}) \quad (4)$$

based on the previous time steps. Lower order extrapolations are used for the first two time steps.

IQN-ILS

The FSI problem in root-finding formulation is a set of nonlinear equations in the interface's position

$$R(\mathbf{x}) = 0 \quad (5)$$

which could be solved by means of Newton-Raphson iterations

$$\text{solve } \left. \frac{dR}{d\mathbf{x}} \right|_{\mathbf{x}^k} \Delta\mathbf{x} = -\mathbf{r}^k \quad (6)$$

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \Delta\mathbf{x} \quad (7)$$

with the residual calculated as

$$\mathbf{r}^k = R(\mathbf{x}^k) = S \circ F(\mathbf{x}^k) - \mathbf{x}^k = \tilde{\mathbf{x}}^{k+1} - \mathbf{x}^k. \quad (8)$$

The Newton-Raphson iterations in the time step have converged when $\|\mathbf{r}^k\|_2$ is smaller than the convergence tolerance ε_0 . However, the exact Jacobian of R is unknown as the Jacobians of F and S are unavailable because F and S represent (commercial) black box codes.

If the Jacobian of R is approximated and quasi-Newton iterations are performed, black-box solvers can be used but this approach does not prevent that the linear system (6) has to be solved. It is more

advantageous to approximate the *inverse* of the Jacobian by applying the least-squares technique introduced by Vierendeels et al. [11] on a particular set of vectors as will be explained below. The interface quasi-Newton iterations with the approximation for the inverse of the Jacobian from a least-squares model (IQN-ILS) [10] can then be written as

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \left(\widehat{\frac{dR}{d\mathbf{x}}} \bigg|_{\mathbf{x}^k} \right)^{-1} (-\mathbf{r}^k) \quad (9)$$

It can be seen from equation (9) that the approximation for the inverse of the Jacobian does not have to be created explicitly; a procedure to calculate the product of this matrix with the vector $\Delta \mathbf{r} = -\mathbf{r}^k$ is sufficient.

Equation (8) shows that the flow equations and structural equations are solved in quasi-Newton iteration k , resulting in $\tilde{\mathbf{x}}^{k+1}$ and the corresponding residual $\mathbf{r}^k$. The difference (Δ) between the vectors $\tilde{\mathbf{x}}^{i+1}$ and $\mathbf{r}^i$ from all previous iterations and the most recent vectors in $\tilde{\mathbf{x}}^{k+1}$ and $\mathbf{r}^k$ is calculated for $i = 0, \dots, k-1$ and these vectors with differences are stored as the columns of the $u \times v$ matrices

$$V^k = [\Delta \mathbf{r}^{k-1} \quad \Delta \mathbf{r}^{k-2} \dots \Delta \mathbf{r}^1 \quad \Delta \mathbf{r}^0] \quad (10)$$

and

$$W^k = [\Delta \tilde{\mathbf{x}}^k \quad \Delta \tilde{\mathbf{x}}^{k-1} \dots \Delta \tilde{\mathbf{x}}^2 \quad \Delta \tilde{\mathbf{x}}^1]. \quad (11)$$

The vector $\Delta \mathbf{r} = \mathbf{0} - \mathbf{r}^k$ is approximated as a linear combination $\Delta \mathbf{r} \approx V^k \mathbf{c}^k$ of the known $\Delta \mathbf{r}^i$ with $\mathbf{c}^k \in \mathbb{R}^v$ the coefficients of the decomposition. As $v \leq u$, the decomposition is an overdetermined set of equations for the elements of $\mathbf{c}^k$ and hence the least-squares solution to this linear system is calculated. For that reason, V^k is decomposed as

$$V^k = Q^k R^k \quad (12)$$

with the so-called economy size QR-decomposition [14]. The coefficient vector $\mathbf{c}^k$ is then determined by solving the triangular system

$$R^k \mathbf{c}^k = Q^{kT} \Delta \mathbf{r}. \quad (13)$$

The $\Delta \tilde{\mathbf{x}}$ that corresponds to $\Delta \mathbf{r}$ is subsequently calculated as a linear combination of the previous $\Delta \tilde{\mathbf{x}}^{i+1}$, giving

$$\Delta \tilde{\mathbf{x}} = W^k \mathbf{c}^k. \quad (14)$$

From equation (8), it follows that $\Delta \mathbf{r} = \Delta \tilde{\mathbf{x}} - \Delta \mathbf{x}$ which is substituted in equation (14) such that

$$\Delta \mathbf{x} = W^k \mathbf{c}^k - \Delta \mathbf{r}. \quad (15)$$

Because the coefficients $\mathbf{c}^k$ are a function of $\Delta \mathbf{r}$, equation (15) shows how $\Delta \mathbf{x}$ can be approximated for a given $\Delta \mathbf{r}$. Hence, equation (15) can be seen as the sought-after procedure to calculate the product of the approximation for the inverse of the Jacobian and a vector $\Delta \mathbf{r}$.

The complete IQN-ILS technique is shown in Figure 1(a). For details like the start-up of the algorithm and extensions like reuse of data from previous time steps, the reader is referred to [10].

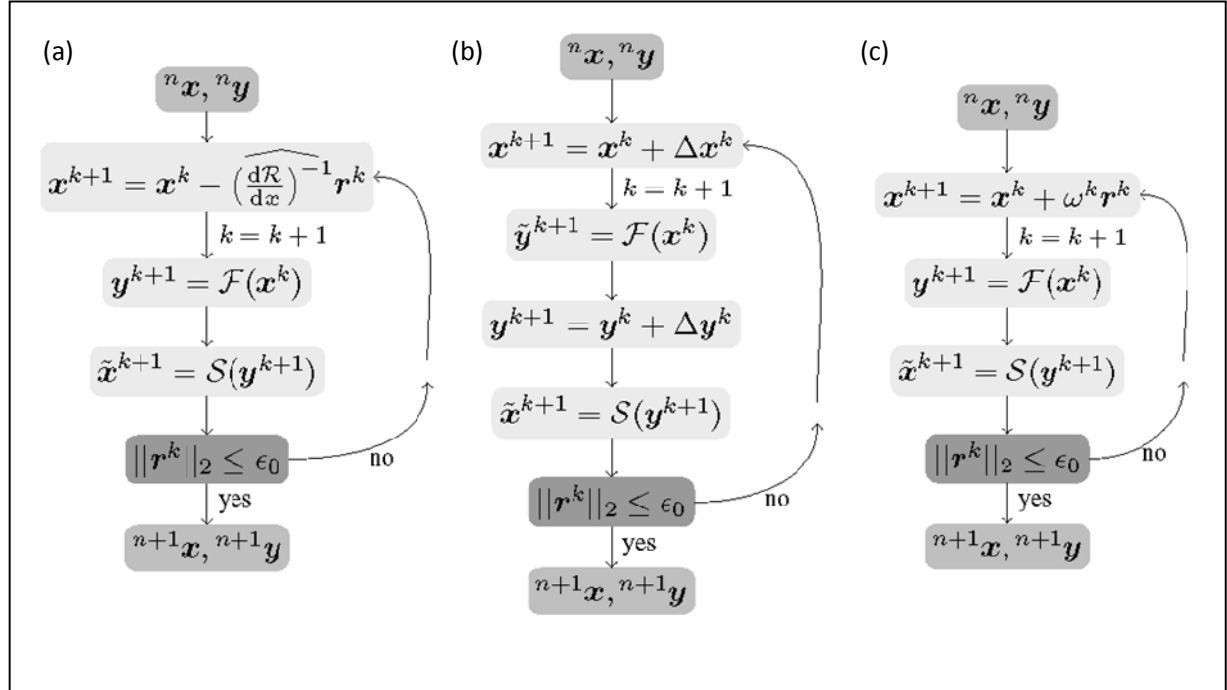


Figure 1: Simplified scheme of (a) the IQN-ILS technique, (b) the IBQN-LS technique and (c) (Aitken) relaxation

IBQN-LS

The interface block quasi-Newton method with approximate Jacobians from least-squares models (IBQN-LS) [11] is depicted in Figure 1(b). This coupling technique solves the FSI problem written as

$$F(x) - y = 0 \quad (16)$$

$$S(y) - x = 0 \quad (17)$$

with block Newton-Raphson iterations of the Gauss-Seidel type using approximate Jacobians for the flow solver and structural solver. The resulting linear system

$$\begin{bmatrix} \widehat{\frac{dF}{dx}} & -I \\ -I & \widehat{\frac{dS}{dy}} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = - \begin{bmatrix} F(x) - y \\ S(y) - x \end{bmatrix} \quad (18)$$

is thus first solved for Δx , followed by an update of x and the right-hand side. Subsequently, the modified system is solved for Δy and y is updated. As a consequence, the IBQN-LS method modifies the

stress distribution that is calculated by the flow solver before transferring it to the structural solver, as opposed to the two other techniques described in this paper.

The linear systems in the block Newton-Raphson iterations are solved with an iterative solver, e.g. the generalized conjugate residual method or the generalized minimal residual method. Hence, only procedures to calculate the product of the approximate Jacobians $\frac{\widehat{dF}}{dx}$ and $\frac{\widehat{dS}}{dy}$ with an arbitrary vector have to be known. For $\frac{\widehat{dF}}{dx}$, this procedure is constructed from the inputs $\mathbf{x}^i$ and outputs $\mathbf{\tilde{y}}^{i+1}$ ($i = 0, \dots, k$) of the flow solver during the coupling iterations in the same way as explained for the IQN-ILS technique. The procedure to calculate the product of $\frac{\widehat{dS}}{dy}$ with a vector is generated analogously from the inputs and outputs of the structural solver.

Aitken relaxation

Aitken relaxation [24], shown in Figure 1(c), determines a dynamically varying scalar relaxation factor ω^k for the fixed-point iterations within a time step.

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \omega^k \mathbf{r}^k \quad (19)$$

The value of the relaxation factor ω^k is obtained as

$$\omega^k = -\omega^{k-1} \frac{(\mathbf{r}^{k-1})^T (\mathbf{r}^k - \mathbf{r}^{k-1})}{(\mathbf{r}^k - \mathbf{r}^{k-1})^T (\mathbf{r}^k - \mathbf{r}^{k-1})}. \quad (20)$$

Equation (19) demonstrates that Aitken relaxation can also be interpreted as an interface quasi-Newton technique: if the inverse of the Jacobian in equation (9) is approximated by $-\omega^k I$, the Aitken relaxation method is retrieved.

Results

Oscillation of a beam

The first example is the unsteady FSI2 test from the benchmark which is described in detail in [12]. The geometry consists of a horizontal channel of 0.41m high, containing a rigid cylinder with centre positioned 0.2m above the bottom of the channel. A laminar, viscous flow with density 10^3kg/m^3 and kinematic viscosity $10^{-3} \text{m}^2/\text{s}$ enters the channel from the left hand side with a parabolic velocity profile and a mean inlet velocity of 1.0m/s. A constant pressure is imposed at the outlet of the channel and a no-slip boundary condition is applied on the walls of the channel. A linearly elastic beam is attached to the right hand side of the cylinder and this beam has a density of 10^4kg/m^3 , a Young's modulus of $1.40 \cdot 10^6 \text{N/m}^2$ and a Poisson's ratio of 0.4.

The flow problem is solved by the finite volume flow solver Fluent (Ansys, Inc.) with second order discretization for the pressure and second order upwind for the momentum. The mesh of the fluid domain is adapted to the position of the fluid-structure interface with a spring analogy and remeshing is

performed if the skewness of the cells is too large. The finite element structural solver Abaqus (Simulia, Inc.) uses implicit time integration of plain strain elements with 8 nodes and takes into account the geometric nonlinearities due to the large deformation of the structure. The coupling algorithm is executed on one processor, the flow solver and structural solver each on two processors.

The number of solver evaluations and the relative duration of the simulations are given in Table 1. The residual $\|r^i\|_2$ is reduced five orders of magnitude with respect to its initial value. The number between brackets behind the name of an algorithm indicates from how many time steps information is reused. Reuse of information from 3 time steps reduces the number of coupling iterations with approximately 30% for both IQN-ILS and IBQN-LS. For this unsteady simulation, the duration of the simulation is similar for IQN-ILS(3) and IBQN-LS(3), which are significantly faster than Aitken relaxation. Pressure contours in the vicinity of the structure are shown in Figure 2.

Table 1: Number of solver evaluations and relative duration for the unsteady FSI2 test with the oscillating flexible beam. The number of solver evaluations per time step has been averaged over the last period of the oscillation.

Algorithm	Evaluations	Duration
IQN-ILS	9.4	1.84
IQN-ILS(3)	6.1	1.07
IBQN-LS	7.2	1.54
IBQN-LS(3)	4.8	1.00
Aitken relaxation	9.9	1.81

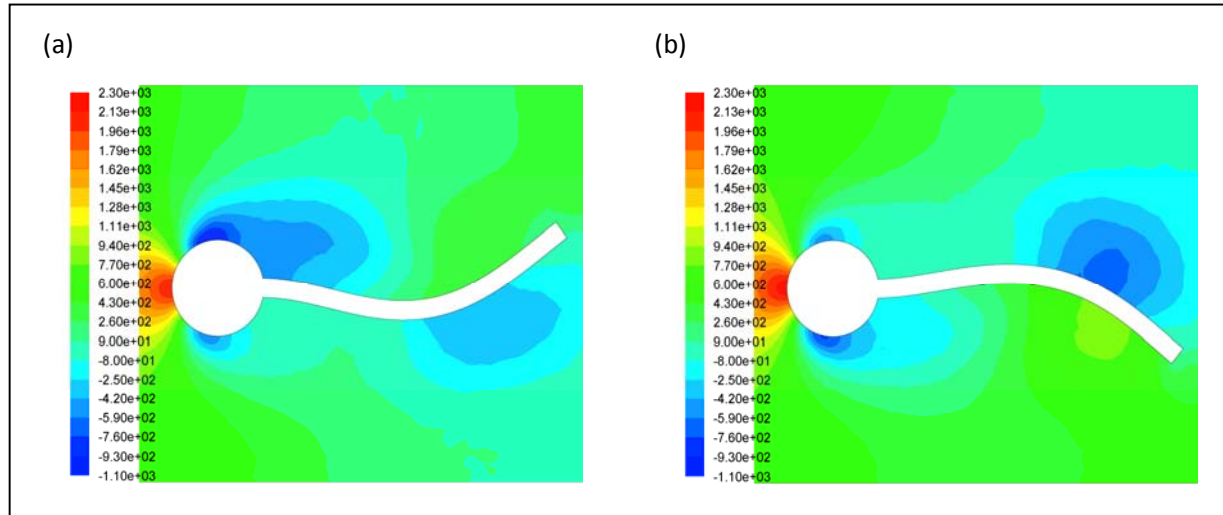


Figure 2: Pressure contours in the 2D unsteady FSI2 test with the oscillating flexible beam after (a) 12s and (b) 16s.

Propagation of a pressure wave in a tube

The second test case is a 3D simulation of a straight flexible tube — representing an artery — with radius 0.005m and length 0.05m, as described by Fernandez et al. [13]. The same flow solver and structural solver as in the previous example have been used, but the structure has been modelled with

one layer of shell elements. The tube's wall is a linear elastic material with density 1200kg/m^3 , Young's modulus $3 \cdot 10^5\text{N/m}^2$, Poisson's ratio 0.3 and thickness 0.001m . The structure is clamped in all directions at the inlet and outlet. The fluid is incompressible and has a density of 1000kg/m^3 and a viscosity of $0.003\text{Pa} \cdot \text{s}$.

Both the fluid and the structure are initially at rest. During the first $3 \cdot 10^{-3}\text{s}$, an overpressure of 1333.2N/m^2 is applied at the inlet. The pressure wave propagates through the tube during 10^{-2}s , simulated with time steps of 10^{-4}s . The fluid model contains 37128 degrees-of-freedom and the structure model 29760. Pressure contours on the fluid-structure interface are shown in Figure 3 and they correspond well with the above mentioned reference.

Table 2 gives the number of solver evaluations in a time step, averaged over the entire simulation, and the relative duration of the simulations. In every time step, $\|r^i\|_2$ is reduced three orders of magnitude with respect to its initial value in that time step. For this case, the duration of the simulation is again similar for IBQN-LS(10) and IQN-ILS(10), which are both faster than Aitken relaxation.

Table 2: Number of solver evaluations per time step and relative duration for the simulation of the propagation of a pressure wave in a 3D flexible tube. The number of solver evaluations per time step has been averaged over the entire simulation.

Algorithm	Evaluations	Duration
IQN-ILS(2)	8.4	1.30
IQN-ILS(10)	6.6	1.03
IBQN-LS(2)	8.2	1.29
IBQN-LS(10)	6.3	1.00
Aitken relaxation	26.7	4.69

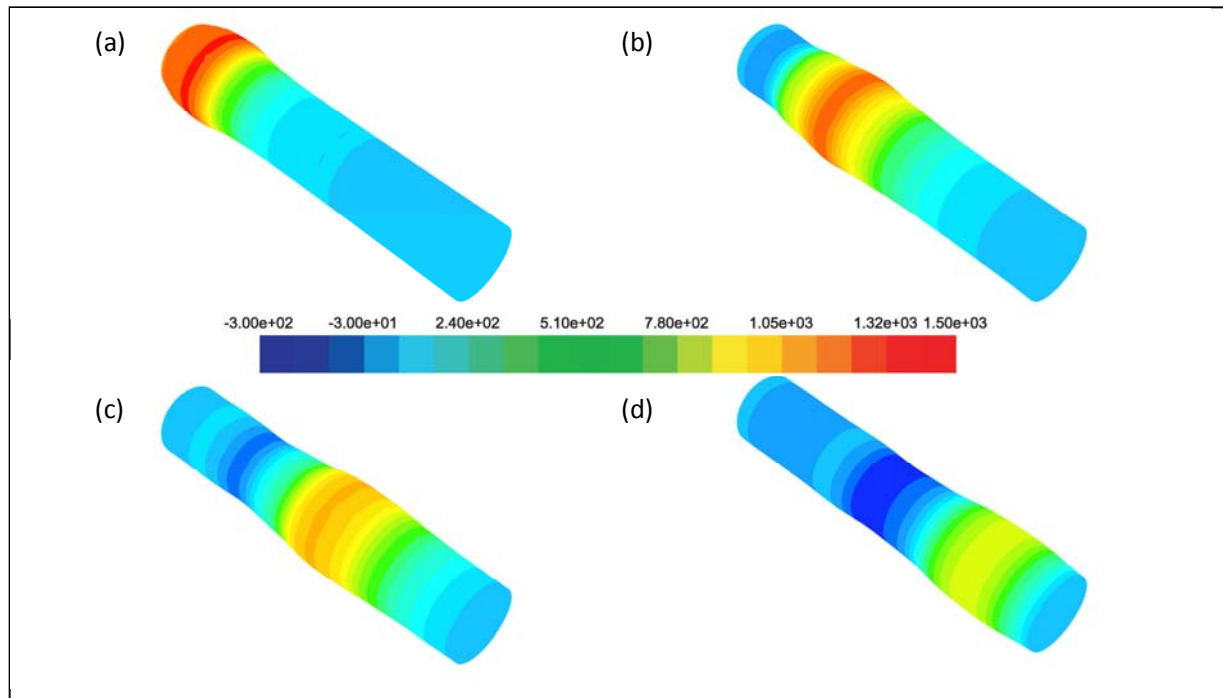


Figure 3: Pressure contours on the fluid-structure interface in a simulation of the propagation of a pressure wave in a 3D flexible tube after (a) $2.5 \cdot 10^{-3}\text{s}$, (b) $5 \cdot 10^{-3}\text{s}$, (c) $7.5 \cdot 10^{-3}\text{s}$ and (d) $10 \cdot 10^{-3}\text{s}$. The deformations have been magnified by a factor 10.

Conclusion

The IQN-ILS technique, the IBQN-LS technique and fixed-point iterations with Aitken relaxation are valuable coupling techniques for the simulation of fluid-structure interaction with incompressible fluids. They use the flow solver and the structural solver as a black box and hence they can be implemented in future versions of MpCCI. The results have demonstrated that the performance of IQN-ILS and IBQN-LS is similar and better than the performance of Aitken relaxation. The implementation of IQN-ILS is nevertheless only slightly more complex than the implementation of Aitken relaxation.

Acknowledgments

The authors gratefully acknowledge the funding of Joris Degroote by a Ph.D. fellowship of the Research Foundation - Flanders (FWO). Jan Vierendeels acknowledges the funding by Research Project G027508 of the Research Foundation - Flanders and the Ghent University Association.

References

- [1] C. Farhat, K. van der Zee, P. Geuzaine, Provably second-order time-accurate loosely coupled solution algorithms for transient nonlinear computational aeroelasticity, *Computer Methods in Applied Mechanics and Engineering* 195 (2006) 1973–2001.
- [2] K. Willcox, J. Paduano, J. Peraire, Low order aerodynamic models for aeroelastic control of turbomachines, in: 40th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials (SDM) Conference, St Louis, MO, 1999, pp. 1–11.
- [3] K. Stein, T. Tezduyar, V. Kumar, S. Sathe, R. Benney, R. Charles, Numerical simulation of soft landing for clusters of cargo parachutes, in: P. Niettaanmäki, T. Rossi, K. Majava, O. Pieronneau (Eds.), *European congress on computational methods in applied sciences and engineering ECCOMAS 2004*, Jyväskylä, 2004, pp. 1–14.
- [4] P. Causin, J.-F. Gerbeau, F. Nobile, Added-mass effect in the design of partitioned algorithms for fluid-structure problems, *Computer Methods in Applied Mechanics and Engineering* 194 (42–44) (2005) 4506–4527.
- [5] N. dos Santos, J.-F. Gerbeau, J.-F. Bourgat, Partitioned FSI strategy for simulations of a thin elastic valve, in: P. Wesseling, E. Oñate, J. Périaux (Eds.), *European Conference on Computational Fluid Dynamics ECCOMAS CFD 2006*, ECCOMAS, Delft, The Netherlands, 2006, pp. 1–10.
- [6] M. Lesoinne, C. Farhat, A higher-order subiteration free staggered algorithm for nonlinear transient aeroelastic problems, *AIAA Journal* 36 (9) (1998) 1754–1756.
- [7] S. Piperno, C. Farhat, B. Larrouturou, Partitioned procedures for the transient solution of coupled aeroelastic problems - Part I: model problem, theory and two-dimensional application, *Computer Methods in Applied Mechanics and Engineering* 124 (1995) 79–112.

- [8] A. van Zuijlen, S. Bosscher, H. Bijl, Two level algorithms for partitioned fluid-structure interaction computations, *Computer Methods in Applied Mechanics and Engineering* 196 (2007) 1458–1470.
- [9] U. Küttler, W. Wall, Fixed-point fluid-structure interaction solvers with dynamic relaxation, *Computational Mechanics* 43 (1) (2008) 61–72, DOI: 10.1007/s00466-008-0255-5.
- [10] J. Degroote, K.-J. Bathe, J. Vierendeels, Performance of a new partitioned procedure versus a monolithic procedure in fluid–structure interaction, *Computers & Structures*, (2009) DOI: 10.1016/j.compstruc.2008.11.013
- [11] J. Vierendeels, L. Lanoye, J. Degroote, P. Verdonck, Implicit coupling of partitioned fluid-structure interaction problems with reduced order models, *Computers & Structures* 85 (11–14) (2007) 970–976.
- [12] S. Turek, J. Hron, Proposal for numerical benchmarking of fluid-structure interaction between an elastic object and laminar incompressible flow, in: H.-J. Bungartz, M. Schäfer (Eds.), *Fluid–Structure Interaction – Modelling, Simulation, Optimisation*, No. 53 in *Lecture Notes in Computational Science and Engineering*, Springer, Berlin, 2006, pp. 371–385, ISBN 3-540-34595-7.
- [13] M. Fernandez, M. Moubachir, A Newton method using exact Jacobians for solving fluid-structure coupling, *Computers & Structures* 83 (2005) 127–142.
- [14] G. H. Golub, C. F. V. Loan, *Matrix computations*, 3rd Edition, Johns Hopkins University Press, Baltimore, MD, USA, 1996.