Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

A FULLY PARTITIONED LAGRANGIAN FRAMEWORK FOR FSI PROBLEMS CHARACTERIZED BY FREE SURFACES, LARGE SOLID DEFORMATIONS AND DISPLACEMENTS, AND STRONG ADDED-MASS EFFECTS

M.L. Cerquaglia

University of Liège, Department of Aerospace and Mechanical Engineering, Allée de la découverte 13a, 4000 Liège, Belgium

D. Thomas

University of Liège, Department of Aerospace and Mechanical Engineering, Allée de la découverte 13a, 4000 Liège, Belgium

R. Boman

University of Liège, Department of Aerospace and Mechanical Engineering, Allée de la découverte 13a, 4000 Liège, Belgium

V. Terrapon

University of Liège, Department of Aerospace and Mechanical Engineering, Allée de la découverte 13a, 4000 Liège, Belgium

J.-P. Ponthot

University of Liège, Department of Aerospace and Mechanical Engineering, Allée de la découverte 13a, 4000 Liège, Belgium

Abstract

In this work a fully partitioned Lagrangian framework for the solution of fluid–structure interaction (FSI) problems involving free surfaces, large solid displacements and deformations, and strong added mass effects is presented. The fluid is simulated using the Particle Finite Element Method (PFEM), while Metafor, a large deformations nonlinear Finite Element code, is employed to simulate the solid part. The fully partitioned coupling is ensured through an Interface Quasi-Newton Inverse Least Squares (IQN-ILS) (Degroote et al., 2009) strategy to avoid added mass effects. The Lagrangian particle nature of the PFEM allows the simulation of problems involving free surfaces and very large solid displacements, usually

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

difficult to achieve with traditional bodyfitted CFD techniques. We show that owing to the generality of its formulation the PFEM can be used *as is* in the framework of fully partitioned FSI coupling schemes, where minimal information (i.e. loads and displacements at the FSI interface) is exchanged between the fluid and the solid solvers. More importantly, we demonstrate that a fully partitioned PFEM–FEM coupling based on the IQN-ILS strategy allows the simulation of a very large spectrum of FSI problems without incurring added-mass instabilities. The performance of the IQN-ILS coupling strategy in a fully Lagrangian framework is also assessed and compared to more traditional approaches such as Block-Gauss–Seidel (BGS) iterations with Aitken relaxation. An extensive work of verification and benchmarking is proposed, aiming to encompass all the combinations of physical and numerical parameters possibly leading to added-mass instabilities, and testing the IQN-ILS strategy on different benchmarks beyond those already proposed in the literature. The coupling is performed through CUPyDO (Thomas et al., 2019), a general Python framework for partitioned FSI coupling.

Keywords: Fluid–structure interaction; Partitioned approaches; CUPyDO; Added mass; Free-surface flows; Particle finite element method

1. Introduction

In the last decades the need for understanding and modeling increasingly complex phenomena led to the development of multiphysics coupled models that, in most cases, demand a numerical solution. Often, each physics involved is already complex by itself and it is difficult to come up with a monolithic numerical formulation that encompasses all the features of the coupled problem. On the contrary, specific efficient and robust solution strategies for each physics usually exist. It is thus more attractive to look for coupling techniques that can take advantage of the capabilities of already existing solvers. This kind of approach characterizes the so-called partitioned coupling schemes, where the coupled solvers are treated as black boxes and exchange only minimal information. Many partitioned strategies have been developed in the framework of FSI problems [1–8], with applications ranging from aerospace to biomedical engineering. The coupling strategy may vary depending on the problem at hand. For instance, partitioned approaches are known to suffer convergence or stability problems in cases where the densities of the solid and the fluid are very close to each other. To describe these cases, the generic term *added-mass effects* is employed in the literature and special techniques have to be developed in such situations.

Alongside with the need for multiphysics simulations, the analysis of complex flows has gained growing attention from both the academia and the industry. The study of free-surface and/or interface flows is of particular interest for ships, offshore and dam engineering, environmental engineering (sediments or pollutants transport in rivers and floods, for instance), biomedical applications, the oil and gas industry, and mold filling processes, to name a few. Traditional body-fitted CFD techniques, based on an Eulerian

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

formalism, are not well suited to take into account evolving surfaces or interfaces, even if techniques like the Volume-of-fluid [9] or level-set methods [10] offer a viable (yet costly) solution to the problem. For the same reasons traditional CFD algorithms cannot be easily employed in FSI problems involving very large solid deformations possibly leading to unacceptable fluid mesh distortions.

This paper presents a fully partitioned Lagrangian framework to solve FSI problems characterized by the presence of free surfaces, large solid deformations and displacements, and strong added mass effects. In particular, the approach relies on the Particle Finite Element Method (PFEM) [11,12] to simulate the fluid, on a large deformations Finite Element (FE) solver to simulate the solid, and on the Interface Quasi-Newton Inverse Least Squares (IQN-ILS) strategy to ensure the coupling between the two and avoid added-mass issues.

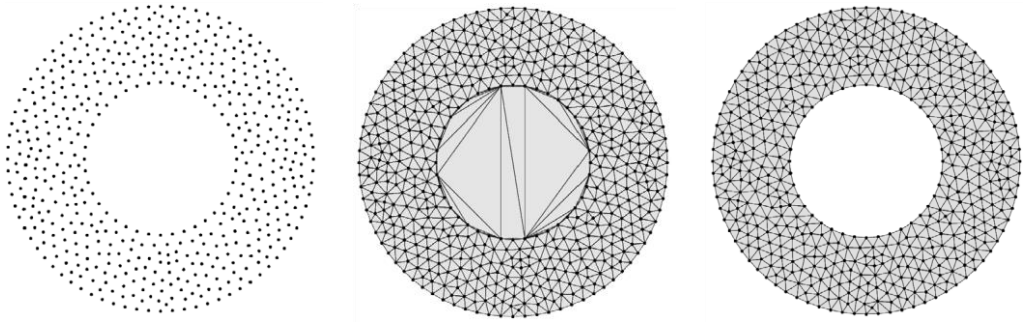
The PFEM is a Lagrangian particle method whose formulation relies on the well-established theory of the Finite Element Method (FEM). Owing to its Lagrangian particle nature, the PFEM can easily and accurately take into account moving boundaries, free surfaces, and evolving interfaces in general. In the context of FSI simulations, these same properties lead to the possibility of simulating problems involving very large solid displacements in a natural way, without the need of any *mesh-morphing* procedure, typical of classical body-fitted CFD approaches. Moreover, unlike other particle or *meshless* methods, such as the Smooth Particle Hydrodynamics (SPH) [13,14] or the Elementfree Galerkin (EFG) [15] methods, the PFEM is very robust with respect to the imposition of essential boundary conditions and to the introduction of complex constitutive models, and its convergence and stability properties derive naturally from those of Finite Elements. The generality of the method makes it very attractive in the perspective of a fully partitioned coupling, since it can be used *as is*, and new constitutive models can be added without the need of specific treatments or consequences on the coupling strategy. However, limited effort has been made in the literature to assess the capabilities of the PFEM in a fully partitioned FSI framework. Idelsohn et al. [16] and Franci et al. [17] proposed monolithic formulations where the fluid and the solid are both modeled using the PFEM. Zhu and Scott [18] presented a semi-partitioned approach, where an incompressible fluid is modeled with the PFEM and the solid with FEM but within the same code so that a unique system of coupled equations is solved, employing a fractional step procedure. Very recently Meduri et al. [19] proposed a fully-explicit partitioned coupling between an in-house PFEM solver (used for the fluid) and Abaqus/Explicit (used for the solid, employing FEM). Unfortunately, their procedure is based on the introduction of Lagrange multipliers at the fluid–solid interface, thus breaking the partitioned nature of the approach since both solvers have to be modified to account for the Lagrange multipliers. Moreover, this approach is only valid for explicit fluid and solid solvers.

In this work we demonstrate that combining the PFEM with an advanced coupling strategy such as the IQN-ILS can be used in a fully partitioned way to solve a very broad spectrum of FSI applications. To this aim, the IQNILS technique is tested against a series of demanding benchmarks, beyond those already

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

presented in the literature, covering large part of the envelope of physical and numerical parameters possibly leading to added-mass effects. To the best of the authors knowledge this is the first time that a fully partitioned PFEM/FEM coupling based on the IQN-ILS technique is proposed, with the aim of solving problems characterized by strong added mass effects. It is also the first time that the IQN-ILS technique is assessed in a fully Lagrangian framework and, in particular, in combination with the PFEM method to solve FSI problems where free surfaces are present.

Fig. 1. Boundary identification in the Particle Finite Element Method (PFEM): α-shapes technique. The correct solution is obtained for α = 1.5.



(a) $\alpha = 0.$ (b) $\alpha = 10$ (c) $\alpha = 1.5$

The IQN-ILS coupling strategy has been implemented in CUPyDO [20], a general framework for solving FSI problems (including both mechanical and thermal coupling), extending the work already performed by the authors in [20]. CUPyDO is based on a fully partitioned solution strategy where each solver is treated as a black box. The only information exchanged between the solvers involves elementary quantities at the fluid–solid interface. The communication is made through memory, based on a Python wrapping strategy, i.e. a Python layer above both solvers. For an additional solver to be plugged into CUPyDO only a minimal Python interface has to be written. Different time integration schemes (implicit and/or explicit) and time steps (including adaptive time stepping) can be independently used by the fluid and solid solvers. Synchronization between the solvers is ensured by CUPyDO based on the FSI time step size chosen by the user. In order to obtain a physically meaningful solution the FSI time step has to be chosen (usually based on dimensional analysis considerations) such that each physics is correctly captured and the different dynamics involved are not filtered by the coupling procedure. Moreover, when a strong coupling procedure is employed, a too large time step would prevent the convergence of the coupled solution.

The paper is organized as follows. In Section 2 the PFEM method is briefly described. In Section 3 the equations governing the coupled FSI problem are derived. Section 4 is dedicated to the description of the coupling algorithms implemented in CUPyDO, with particular emphasis on added-mass effects and on the techniques employed to circumvent them, in particular the IQN-ILS strategy. The performance of the present framework for problems involving free-surfaces, large solid deformations and displacements, and

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

strong added-mass effects is analyzed in Section 5 on a series of demanding benchmarks. Conclusions are drawn in Section 6.

2. The Particle Finite Element Method (PFEM)

In this work the fluid is simulated using the PFEM. The PFEM is a fairly recent numerical approach that combines the advantages of the FEM with an efficient remeshing and boundary recognition strategy. In the PFEM, the fluid is discretized using a set of points, hereafter referred to as particles, which are Lagrangian entities that actually represent material points of the body. In order to evaluate the forces acting on each particle, a new mesh is built at each time step from the entire set of particles using a Delaunay triangulation. The boundaries of the domain are then identified using a geometric criterion based on the α-shapes technique [21]. According to the α-shapes criterion, triangles whose circumcircle radius exceeds a given representative length, scaled by a parameter α, are deleted. In this way, depending on the value of α, too large or too distorted triangles can be eliminated from the triangulation, as depicted in Fig. 1. A drawback of the use of α-shapes is that it can induce non physical volume variations. To limit this phenomenon a value of α between 1.2 and 1.5 should be employed, as discussed by Franci and Cremonesi [22]. Delaunay triangulation combined with α-shapes allows a fast evaluation of nodal connectivity even when the total number of particles becomes very large [23]. Classical finite element shape functions can then be defined on the new mesh to solve the corresponding weak form of the governing equations. Linear shape functions are usually employed in the PFEM for all the discretized quantities.

Given a set of particles, the main PFEM steps within one time increment can be summarized as follows:

- Define the particle connectivity through a Delaunay triangulation.
- Identify the domain boundaries using the α-shapes technique.
- Solve the governing equations making use of FE shape functions.
- Use the solution obtained from the previous step to update the particle positions.

In principle the PFEM can be employed to model both solids and fluids. Indeed, it has already been successfully employed to simulate free-surface flows, both compressible [24] and incompressible [11], interface flows [25,26], granular/non-Newtonian fluid flows [27,28], FSI problems [16,17,29,30], forming processes [31], and melting of polymers [32], among others. Finally, it is also worth noting that very recently a new version of the PFEM, called PFEM2 [33] and based on the combined use of particles and a background mesh, has been developed and applied to multi-fluid flows [34] and FSI problems [35]. The main difficulty in the PFEM2, however, resides in the way quantities are transferred from the background mesh to the particles and conversely.

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

In this work the PFEM, in its original form, is mainly employed to simulate free-surface flows involving Newtonian incompressible fluids. In the following, several aspects of the implementation are omitted. For a more detailed description of the method, the reader is referred, for example, to Idelsohn et al. [11] or Oñate et al. [12].

3. Governing equations

This section introduces the mathematical and numerical models for the fluid and solid domains. In particular, the incompressible flow solver based on the PFEM [36] and the nonlinear structural solver Metafor [37] are briefly described. Finally, the coupling conditions at the fluid–solid interface are summarized.

3.1. LAGRANGIAN VS EULERIAN DESCRIPTION OF MOTION

A brief recall of material and spatial coordinates and reference and current configuration is given. This is motivated by the Lagrangian formalism used for the fluid in this work, which is not a common choice in CFD. In the following the definitions and notations of Belytschko et al. [38] are employed.

Spatial coordinates are denoted by $\boldsymbol{x}$. A spatial coordinate specifies the location of a point in space. Material coordinates are denoted by $\boldsymbol{X}$. The material coordinate labels a material point: each material point has a unique material coordinate, which is usually taken to be its spatial coordinate in the initial configuration, i.e. $\boldsymbol{X} = \boldsymbol{x}$ at $t = 0$. The collection of material points with coordinates X defines the *reference configuration*, while the collection of material points with coordinates $\boldsymbol{x}$ defines the *current configuration*.

In a Lagrangian description, the motion or deformation of a body is described by a function $\boldsymbol{\phi}(\boldsymbol{X}, t)$ where the material coordinates $\boldsymbol{X}$ and time t are the independent variables. This function gives the spatial positions of the material points as a function of time through

$$\boldsymbol{x} = \boldsymbol{\phi}(\boldsymbol{X}, t). \tag{1}$$

Conversely, in an Eulerian description, the motion or deformation of a body is described by a function $\boldsymbol{\phi}^{-1}(\boldsymbol{x}, t)$ with the spatial coordinates $\boldsymbol{x}$ and time t as the independent variables. This function gives the material coordinates of the point which is at position $\boldsymbol{x}$ at time t through

$$\boldsymbol{X} = \boldsymbol{\phi}^{-1}(\boldsymbol{x}, t). \tag{2}$$

In this work the governing equations are written in the current configuration and a Lagrangian description of motion is used.

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

3.2. NEWTONIAN INCOMPRESSIBLE FLUID

At a given time t, and in the current configuration $\boldsymbol{x}$, the momentum and mass conservation equations for a Newtonian incompressible fluid read:

$$\rho\frac{D\boldsymbol{v}}{Dt} = -\nabla p + \mu\nabla\cdot\left(\nabla\boldsymbol{v} + \nabla\boldsymbol{v}^{\mathrm{T}}\right) + \rho\mathbf{b} \quad \text{in} \quad \Omega_f, \tag{3}$$

$$\nabla\cdot\boldsymbol{v} = 0 \quad \text{in} \quad \Omega_f, \tag{4}$$

where $\frac{D(\cdot)}{Dt}$ is the Lagrangian, or material, time derivative of the quantity $(\cdot)$, Ω_f is the volume occupied by the fluid in the current configuration, $\rho(\boldsymbol{x},t)$ is the current density, μ is the dynamic viscosity, $\boldsymbol{v} = \boldsymbol{v}(\boldsymbol{x},t)$ is the velocity field, $p(\boldsymbol{x},t)$ is the pressure, and $\rho\,\boldsymbol{b}(\boldsymbol{x},t)$ is some body force, such as gravity for instance. Notice that since the momentum conservation equation is written in Lagrangian form the nonlinear convective term is not explicitly present.

These equations have to be complemented with Dirichlet and Neumann boundary conditions:

$$\boldsymbol{v}(\boldsymbol{x},t) = \bar{\boldsymbol{v}}(\boldsymbol{x},t) \quad \forall\boldsymbol{x}\in\Gamma_D^f, \tag{5}$$

$$\boldsymbol{\sigma}(\boldsymbol{x},t)\cdot\mathbf{n} = \bar{\mathbf{t}}(\boldsymbol{x},t) \quad \forall\boldsymbol{x}\in\Gamma_N^f, \tag{6}$$

where $\bar{\boldsymbol{v}}(\boldsymbol{x},t)$ and $\bar{\boldsymbol{t}}(\boldsymbol{x},t)$ are imposed velocities and surface tractions respectively, $\boldsymbol{\sigma} = \boldsymbol{\sigma}(\boldsymbol{x},t)$ is the Cauchy stress tensor, $\boldsymbol{n}$ denotes the unit outward normal to the boundary, $\Gamma_D^f \cup \Gamma_N^f = \partial\Omega_f$, where $\partial\Omega_f$ is the boundary of the current volume Ω_f, and $\Gamma_D^f \cap \Gamma_N^f = \emptyset$.

By introducing a *linear* isoparametric finite element discretization for both velocity and pressure, the following system of semi-discrete equations is obtained:

$$\mathbf{M}_f\frac{D\mathbf{v}}{Dt} + \mathbf{K}_f\mathbf{v} + \mathbf{G}_f\mathbf{p} = \mathbf{f}, \tag{7}$$

$$-\mathbf{G}_f^{\mathrm{T}}\mathbf{v} + \mathbf{S}\mathbf{p} = \mathbf{0}, \tag{8}$$

where $\mathbf{v}$ and $\mathbf{p}$ are the discretized form of the velocity and pressure fields respectively, $\mathbf{M}_f$ is the mass matrix, $\mathbf{K}_f$ the matrix containing the viscous terms, and $\mathbf{G}_f$ the discrete version of the gradient operator. The subscript f refers to the fluid. In the continuity equation, a stabilization matrix $\mathbf{S}$ is introduced since the selected discretization, i.e. linear velocity and pressure, is known to violate the Ladyzhenskaya–Babuška–Brezzi (LBB) condition [39,40]. For the inhouse PFEM solver used in this work a Pressure-Stabilizing Petrov–Galerkin (PSPG) stabilization is used [36,41,42]. This stabilized monolithic approach also guarantees very good volume conservation properties in the case of free surface flows compared to

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

fractional-step schemes [43,44]. Notice that no stabilization is needed for the velocity thanks to the use of a Lagrangian formulation.

The vector $\mathbf{f}$ contains the contribution of both body forces and surface tractions, i.e. the applied forces. Moreover, $\mathbf{f}$ contains surface tension contributions, if present. Surface tension effects can be introduced as concentrated nodal loads $\mathbf{t}_i^{st}$ normal to the free surface, through a simple model:

$$\mathbf{t}_i^{st} = \gamma k \mathbf{n}_i, \tag{9}$$

where $\mathbf{n}_i$ is the local unit normal at node i, γ is the surface tension coefficient and $k\ (\boldsymbol{x}, t)$ is the local curvature.

Eqs.(7) and (8) form a nonlinear system of equations which is solved iteratively using a Picard algorithm. Time integration is performed through a Backward-Euler scheme.

3.3. SOLID EQUATIONS

At a given time t, and in the current configuration $\boldsymbol{x}$, the momentum and mass conservation equations for a solid read:

$$\rho \frac{D^2 \boldsymbol{x}}{Dt^2} - \nabla \cdot \boldsymbol{\sigma} = \rho \mathbf{b} \quad \text{in} \quad \Omega_s, \tag{10}$$

$$\rho J = \rho_0 \quad \text{in} \quad \Omega_s, \tag{11}$$

where Ω_s is the volume occupied by the solid in the current configuration, $\rho\ (\boldsymbol{x}, t)$ and $\rho_0(\boldsymbol{x}, t)$ are the densities in the current and reference configuration respectively, J is the determinant of the deformation gradient tensor $\left(\boldsymbol{F} = \frac{\partial x}{\partial X}\right)$, $\boldsymbol{\sigma}\ (\boldsymbol{x}, t)$ is the Cauchy stress tensor and $\rho \boldsymbol{b}(\boldsymbol{x}, t)$ is some body force. These equations have to be complemented with Dirichlet and Neumann boundary conditions:

$$\boldsymbol{x}\ (t) = \bar{\boldsymbol{x}}\ (t) \quad \forall \boldsymbol{x} \in \Gamma_D^s, \tag{12}$$

$$\boldsymbol{\sigma}\ (\boldsymbol{x}, t) \cdot \mathbf{n} = \bar{\mathbf{t}}\ (\boldsymbol{x}, t) \quad \forall \boldsymbol{x} \in \Gamma_N^s, \tag{13}$$

where $\overline{\boldsymbol{x}}\ (t)$ and $\bar{\boldsymbol{t}}(\boldsymbol{x}, t)$ are imposed displacements and surface tractions respectively, $\mathbf{n}$ denotes the unit outward normal to the boundary, , $\Gamma_D^S \cup \Gamma_N^S = \partial \Omega_s$, where $\partial \Omega_s$ is the boundary of the current volume Ω_s, and , $\Gamma_D^S \cap \Gamma_N^S = \emptyset$.

The in-house nonlinear solver Metafor [37,45] used in this work employs a finite element discretization to solve the above system of equations. The discretized form of Eqs. (10) and (11) is

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

$$\mathbf{M}_s \frac{D^2\mathbf{x}}{Dt^2} + \mathbf{f}_{\text{int}}(\mathbf{d}) = \mathbf{f}_{\text{ext}}, \tag{14}$$

$$\mathbf{M}_s = \mathbf{M}_s^0, \tag{15}$$

where $\mathbf{x}$ is the discretized position vector, $\mathbf{M}_s$ is the mass matrix, and $\mathbf{f}_{\text{int}}$ and $\mathbf{f}_{\text{ext}}$ are the vectors of internal and external forces, respectively.

The discretized mass conservation equation (15) is automatically satisfied since the mass matrix remains constant during the simulation. System (14) is nonlinear and it is solved using a Newton–Raphson strategy as:

$$\mathbf{M}_s \frac{D^2\mathbf{x}}{Dt^2} + \mathbf{K}_t(\mathbf{x}^k)\,\Delta\mathbf{x}^k = \mathbf{f}_{\text{ext}} - \mathbf{f}_{\text{int}}(\mathbf{x}^k) \tag{16}$$

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \Delta\mathbf{x}^k \tag{17}$$

where $\mathbf{K}_t(\mathbf{x}^k)$ is the consistent tangent stiffness matrix defined as $\mathbf{K}_t(\mathbf{x}^k) = \frac{\partial\,\mathbf{f}_{\text{int}}(\mathrm{d})}{\partial \mathrm{d}}|_k$. Time integration is performed employing the Generalized-α method [46].

Metafor is equipped with full nonlinear capabilities, including 2D/3D elements for large strain analysis (Selective Reduced Integration (SRI) [47] and Enhanced Assumed Strain (EAS) [48–50] elements, among others), thermomechanical coupling, frictional contact between deformable bodies or CAD surfaces, Arbitrary Lagrangian Eulerian (ALE) [51,52] formalism, and a large set of constitutive laws (thermo-elasto-visco-plastic, damage, composites, etc.) [53–55] and frictional and tribological models [56].

3.4. COUPLING CONDITIONS

Coupling conditions at the fluid–solid interface Γ are defined by the continuity of displacements $\mathbf{d}^\Gamma$ and loads $\mathbf{t}^\Gamma$:

$$\mathbf{d}_f^\Gamma = \mathbf{d}_s^\Gamma \tag{18}$$

$$\mathbf{t}_f^\Gamma = -\mathbf{t}_s^\Gamma, \tag{19}$$

where the load on the fluid side is given by $\mathbf{t}_f^\Gamma = \boldsymbol{\sigma}_f \mathbf{n}_f$ and the load on the solid side by $\mathbf{t}_s^\Gamma = \boldsymbol{\sigma}_s \mathbf{n}_s$ with $\boldsymbol{\sigma}$ the Cauchy stress tensor. Indices f and s denote the fluid and solid domains, respectively. The normal unit vectors $\mathbf{n}_f$ and $\mathbf{n}_s$ are both pointing outwards from their respective domains.

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

4. Partitioned FSI coupling techniques

In FSI partitioned approaches the equations of each physics, as presented in Section 3, are solved separately by dedicated solvers. The aim of any partitioned strategy is to couple the different physics by exchanging minimal information between the solvers, yet ensuring accuracy and stability of the coupled solution.

Among the different existing possibilities, CUPyDO [20] employs the so-called Dirichlet–Neumann [6,57–59] procedure for coupling fluid and solid mechanics equations. For a given fluid interface displacement, $\mathbf{d}_f^\Gamma$, a general nonlinear *Dirichlet* operator $\mathcal{F}$ can be applied to obtain the forces at the fluid interface

$$\mathbf{t}_f^\Gamma = \mathcal{F}\left(\mathbf{d}_f^\Gamma\right). \tag{20}$$

In the same way, a *Neumann* nonlinear operator $\mathcal{S}$ that links the forces applied at the solid interface to the solid interface displacements can be defined as well:

$$\mathbf{d}_s^\Gamma = \mathcal{S}\left(\mathbf{t}_s^\Gamma\right). \tag{21}$$

The coupling conditions expressed by Eqs. (18)–(19) can thus be reformulated as a fixed-point problem [6,59–61]:

$$\mathbf{d}^\Gamma = \mathcal{S}\left(-\mathcal{F}\left(\mathbf{d}^\Gamma\right)\right), \tag{22}$$

where $\mathbf{d}^\Gamma$ is the displacement common to both the solid and fluid interfaces.

In this work, the operators $\mathcal{F}$ and $\mathcal{S}$ represent the fluid and solid solvers, respectively. This notation emphasizes the partitioned nature of CUPyDO, as it treats each individual solver as a blackbox.

Different techniques exist to solve Eq. (22). Before analyzing them in detail the fixed-point problem is first rewritten in a more concise way as:

$$\mathbf{d}^\Gamma = \mathcal{S} \circ \mathcal{F}\left(\mathbf{d}^\Gamma\right) = \mathcal{T}\left(\mathbf{d}^\Gamma\right), \tag{23}$$

where $\mathcal{T}$ is a global nonlinear transfer operator. In practice, problem (23) is condensed on the solid side, since the number of degrees of freedom of the solid interface is usually much smaller than the number of degrees of freedom of the fluid interface. In the case of non-matching interfaces, CUPyDO employs radial-basis functions (RBF) [62,63] in an energy conservative way to map quantities between the solid and the fluid interface [20]. When non-matching interfaces are considered, the quality of the coupled FSI solution depends also on the quality of the interpolation between the fluid and solid data at the common interface.

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

RBF interpolation has proven to be very robust with respect to significant discretization mismatching [63], and for this reason is employed in CUPyDO.

From a practical point of view, problem (23) can thus be rewritten more precisely as

$$\mathbf{d}_s^\Gamma = \mathcal{T}\left(\mathbf{d}_s^\Gamma\right), \tag{24}$$

where the subscript s refers to the solid. In the literature, the term $\mathcal{T}\left(\mathbf{d}_s^\Gamma\right)$ is also often denoted $\tilde{\mathbf{d}}_s^\Gamma$, meaning the displacements computed by the solid solver.

4.1. COUPLING ALGORITHMS IMPLEMENTED IN CUPYDO

In the following, the methods implemented in CUPyDO to solve problem (24) are presented. First, the explicit and Block Gauss–Seidel algorithms already presented in [20] are re-derived for completeness. Then, the newly implemented Interface Quasi-Newton Inverse Least Square (IQN-ILS) strategy [59,64] is presented. Hereafter, the superscript Γ and the subscript s are omitted for clarity, while the superscript k defines the FSI iteration number.

Explicit algorithm. When an explicit coupling is employed, problem (24) becomes simply

$$\mathbf{d}^1 = \mathcal{T}\left(\mathbf{d}^0\right) = \tilde{\mathbf{d}}^1 \tag{25}$$

Block Gauss–Seidel (BGS) algorithm. Problem (24) can also be solved implicitly. In this case, the simplest algorithm is a Block Gauss–Seidel [65–67]:

$$\mathbf{d}^{k+1} = \mathcal{T}\left(\mathbf{d}^k\right) = \tilde{\mathbf{d}}^k, \text{ until } \|\mathbf{r}^k\| = \|\tilde{\mathbf{d}}^k - \mathbf{d}^k\| < \varepsilon_{\text{FSI}}. \tag{26}$$

The residual $\mathbf{r} = \mathcal{T}(\mathbf{d}) - \mathbf{d} = \tilde{\mathbf{d}} - \mathbf{d}$ is defined in the same way for all the implicit coupling schemes implemented in CUPyDO, while ε_{FSI} is a given *dimensional* tolerance chosen by the user.

Block Gauss–Seidel (BGS) algorithm with relaxation. In some cases it is very difficult, or even impossible, to ensure convergence for problem (26). This is typically the case when the solid-to-fluid density ratio is close to or lower than one. This issue is known as added-mass effects, and will be addressed further in the paper. To enhance or even ensure convergence, relaxation can be introduced in Eq. (26) [6]. The problem then becomes

$$\mathbf{d}^{k+1} = \mathcal{T}\left(\mathbf{d}^k\right) + \left(\omega^k - 1\right)\left(\mathcal{T}\left(\mathbf{d}^k\right) - \mathbf{d}^k\right) \tag{27}$$

$$= \tilde{\mathbf{d}}^k + \left(\omega^k - 1\right)\left(\tilde{\mathbf{d}}^k - \mathbf{d}^k\right), \text{ until } \|\mathbf{r}^k\| = \|\tilde{\mathbf{d}}^k - \mathbf{d}^k\| < \varepsilon_{\text{FSI}}. \tag{28}$$

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

The relaxation parameter ω can be constant (static relaxation), or vary from one iteration to another (dynamic relaxation). In the case of dynamic relaxation the most common strategy to update ω is Aitken relaxation [6,68]:

$$\omega^k = -\omega^{k-1} \frac{\mathbf{r}^{k\mathrm{T}} \left(\mathbf{r}^k - \mathbf{r}^{k-1}\right)}{\left|\left|\mathbf{r}^k - \mathbf{r}^{k-1}\right|\right|^2}.$$

The parameter ω can take values between 0 (the displacements do not evolve any longer) and 1 (no relaxation). Usually, an upper limit $\omega_{\max} 1$ is imposed by the user to avoid possible divergence. BGS coupling with Aitken relaxation is in most of the cases successful in solving strongly coupled problems. The capabilities of the BGS coupling with Aitken relaxation implemented in CUPyDO have been assessed by the authors in a previous work [20].

Interface quasi-Newton Inverse Least Square (IQN-ILS) algorithm. Problem (24) can also be addressed using Newton iterations. In this case the solution strategy becomes

$$\mathbf{d}^{k+1} = \mathbf{d}^k + \Delta\mathbf{d}^k, \tag{29}$$

$$\Delta\mathbf{d}^k = -\mathbf{J}^{k^{-1}}\mathbf{r}^k \tag{30}$$

where $\mathbf{J}^k = \frac{\partial \mathbf{r}^k}{\partial \mathbf{d}^k}$ is the Jacobian of the transformation at iteration k. Unfortunately, this Jacobian cannot be readily obtained when using blackbox solvers so that a Quasi-Newton approach must be followed, in which an approximation of $\mathbf{J}$ (or, better, of its inverse) is sought. Eq. (29) becomes then

$$\mathbf{d}^{k+1} = \mathbf{d}^k - \widehat{\mathbf{J}^{k^{-1}}}\mathbf{r}^k, \tag{31}$$

where $\widehat{\mathbf{J}^{k^{-1}}}$ is an approximation of the inverse Jacobian. Several strategies exist to construct this approximation. In the following the Interface Quasi-Newton Inverse Least Square (IQN-ILS) technique proposed by Degroote et al. [59,64,69], and implemented in CUPyDO, will be described in detail. The main advantage of the IQN-ILS approach is that it is matrix-free and directly provides an approximation of the inverse Jacobian, so that no matrix inversion has to be performed and no large matrices have to be stored in memory. In particular, an approximation of the whole term $-\mathbf{J}^{k^{-1}}\mathbf{r}^k$ is built:

$$-\widehat{\mathbf{J}^{k^{-1}}}\mathbf{r}^k \simeq -\frac{\partial \mathbf{d}^k}{\partial \mathbf{r}^k}\mathbf{r}^k. \tag{32}$$

By definition of the residual, $\mathbf{r}^k = \tilde{\mathbf{d}}^k - \mathbf{d}^k$:

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

$$\mathbf{d}^k = \tilde{\mathbf{d}}^k - \mathbf{r}^k \implies -\frac{\partial \mathbf{d}^k}{\partial \mathbf{r}^k}\mathbf{r}^k = \underbrace{\frac{\partial \tilde{\mathbf{d}}^k}{\partial \mathbf{r}^k}\left(-\mathbf{r}^k\right)}_{=\Delta\tilde{\mathbf{d}}^k} + \mathbf{r}^k. \tag{33}$$

By defining $\Delta\mathbf{r}^k = 0 - \mathbf{r}^k$, 0 being the final desired residual, from Eq. (33) one can write:

$$\Delta\tilde{\mathbf{d}}^k = \frac{\partial \tilde{\mathbf{d}}^k}{\partial \mathbf{r}^k}\left(\Delta\mathbf{r}^k\right). \tag{34}$$

Thus, the mapping between $\Delta\mathbf{r}^k$ and $\Delta\tilde{\mathbf{d}}^k$ is needed. Instead of computing this mapping explicitly, successive approximations of $\Delta\mathbf{r}^k$ and $\Delta\tilde{\mathbf{d}}^k$ are built based on information from previous iterations, following the idea of Vierendeels et al. [60]:

$$\Delta\mathbf{r}^k \simeq \sum_{i=0}^{k-1} c_i^k \Delta\mathbf{r}^i = \mathbf{V}^k\mathbf{c}^k, \tag{35}$$

$$\Delta\tilde{\mathbf{d}}^k \simeq \sum_{i=0}^{k-1} c_i^k \Delta\tilde{\mathbf{d}}^i = \mathbf{W}^k\mathbf{c}^k, \tag{36}$$

where

$$\Delta\mathbf{r}^i = \mathbf{r}^i - \mathbf{r}^k, \tag{37}$$

$$\Delta\tilde{\mathbf{d}}^i = \tilde{\mathbf{d}}^i - \tilde{\mathbf{d}}^k, \tag{38}$$

and $\mathbf{V}^k$ and $\mathbf{W}^k$ are matrices collecting differences in $\mathbf{r}^k$ and $\tilde{\mathbf{d}}$:

$$\mathbf{V}^k = \left[\Delta\mathbf{r}^{k-1}\ \Delta\mathbf{r}^{k-2}\ \ldots\ \Delta\mathbf{r}^1\ \Delta\mathbf{r}^0\right], \tag{39}$$

$$\mathbf{W}^k = \left[\Delta\tilde{\mathbf{d}}^{k-1}\ \Delta\tilde{\mathbf{d}}^{k-2}\ \ldots\ \Delta\tilde{\mathbf{d}}^1\ \Delta\tilde{\mathbf{d}}^0\right]. \tag{40}$$

Combining Eqs. (31), (33), and (36) the final expression for $\mathbf{d}^{k+1}$ is

$$\mathbf{d}^{k+1} = \mathbf{d}^k + \Delta\tilde{\mathbf{d}}^k + \mathbf{r}^k = \mathbf{d}^k + \mathbf{W}^k\mathbf{c}^k + \mathbf{r}^k. \tag{41}$$

The remaining unknown coefficient vector $\mathbf{c}^k$ can be calculated from Eq. (35). Recalling that the desired residual should be zero, so that $\Delta\mathbf{r}^k = -\mathbf{r}^k$, Eq. (35) reads

$$\mathbf{V}^k\mathbf{c}^k = -\mathbf{r}^k. \tag{42}$$

This is usually a small problem, since its typical size is the current number of FSI iterations. Nonetheless, it is an overdetermined problem, since the number of FSI iterations, ℓ, is usually much smaller than the number of degrees of freedom on the solid interface, m. Eq. (42) is thus solved in a least square sense,

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

performing an economy-size QR decomposition of matrix $\mathbf{V}^k$, and then computing $\mathbf{c}^k$ by back-substitution:

$$\mathbf{R}^k\mathbf{c}^k = \mathbf{Q}^{k^{\mathrm{T}}}\left(-\mathbf{r}^k\right), \tag{43}$$

where $\mathbf{Q}^k \in \mathbb{R}^{m\times\ell}$ and $\mathbf{R}^k \in \mathbb{R}^{\ell\times\ell}$.

To improve the convergence properties of the IQN-ILS coupling, information from previous time steps can also be retained in matrices V and W. To indicate that information from n previous time steps is retained the expression IQN-ILS(n) will be used in the following.

In order to build matrices $\mathbf{V}$ and $\mathbf{W}$ the existence of at least one previous FSI iteration is required. To circumvent this issue, one iteration of BGS with static relaxation is performed at the beginning of each time step to initialize the IQN-ILS procedure. This problem does not appear when information from previous time steps is used, except at time t = 0, i.e. at the very beginning of the simulation.

Because the vectors used to construct $\mathbf{V}^k$ do not form an orthogonal basis, two or more columns of $\mathbf{V}^k$ can be (almost) linear combinations of one another, especially when information from previous time steps is retained, which can degrade the convergence properties of the method. Some filtering is usually introduced to solve this issue. Two filtering techniques are implemented in CUPyDO: the one suggested by Degroote et al. [64] and referred to as *Degroote* and the one introduced by Haelterman et al. [70] and referred to as *Haelterman*. The basic ideas of the two approaches are briefly summarized here, more details can be found in the corresponding references.

Degroote filtering. In *Degroote* filtering, each time that a diagonal element of the $\mathbf{R}^k$ matrix is smaller (in absolute value) than a given tolerance, the corresponding column of the $\mathbf{V}^k$ matrix is deleted and the QR decomposition is repeated, until all diagonal elements that are too small are eliminated. Thus, a column is deleted if it is small with respect to the others (i.e. the change between two iterations is small). This approach is numerical in nature, since the tolerance employed depends on the accuracy of the fluid and solid solvers.

Haelterman filtering. In *Haelterman* filtering, a column of the $\mathbf{V}^k$ matrix corresponding to a given iteration is deleted if it is not 'sufficiently orthogonal' to all the columns corresponding to previous iterations, i.e. if the norm of the projection of the given column onto all the 'previous' ones is smaller than a given tolerance. However, in this case the tolerance does not depend on the accuracy of the solvers but represents an arbitrary measure of the degree of linear dependency that can be accepted among the columns of $\mathbf{V}^k$. The approach is thus more of an algebraic nature.

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

4.2. ADDED-MASS EFFECTS

In fluid mechanics the term *added-mass* traditionally refers to the virtual additional inertia experienced by a body moving inside a fluid as a consequence of the fact that, by its motion, the body displaces some volume of surrounding fluid. However, more recently the same term has been also employed in the literature to describe numerical instabilities that may appear in FSI problems under certain conditions.

The problem of added-mass effects is typical of partitioned approaches and has been already analyzed in detail in the literature, for instance by Causin et al. [58], Förster et al. [71], Kuttler and Wall [6], Degroote et al. [69,72], Idelsohn et al. [30]. According to these studies, the conditions favorable to the appearance of added-mass instabilities can be summarized to:

- solid-to-fluid density ratio close to or lower than one;
- presence of slender flexible structures;
- small time step size.

While explicit coupling and BGS coupling without relaxation are not suited for problems possibly affected by added-mass effects, BGS with relaxation and IQN-ILS techniques have been developed precisely with the purpose of avoiding instabilities due to added-mass effects.

Finally, it is interesting to mention that Idelsohn et al. [30] observed the same added-mass instabilities in monolithic FSI schemes based on a pressure segregation approach (i.e. employing a fractional step scheme) typically used in presence of incompressible fluids. According to these authors this is due to the fact that a fractional step scheme to solve the fluid equations is equivalent to a staggered solution of the coupled problem, thus incurring the same problems of a proper partitioned strategy.

5. Results

In this section a series of numerical tests involving free surfaces and strong added-mass effects are solved using CUPyDO, coupling Metafor and an in-house PFEM solver. In particular, the performance of the IQN-ILS coupling strategy in this context is analyzed. After a validation benchmark, i.e. the vortex-induced vibrations of a cantilever (Section 5.1), many combinations of physical and numerical parameters possibly leading to added-mass effects are tested:

- *Flexible solid with density close to the one of the fluid*: collapse of a column of fluid against an elastic–plastic obstacle (Section 5.2), oscillations of a column of fluid lying on an elastic solid (Section 5.3), and collapse of a column of fluid through an elastic gate (Section 5.4);

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

- *Almost rigid solid with very low solid-to-fluid density ratio*: pop-off of a very light cylinder initially immersed in a fluid (Section 5.5);
- *Very flexible solid with very low solid-to-fluid density ratio*: filling of an elastic container with some fluid (Section 5.6).

Please notice that, even if the physical nature of some problems is similar, from a numerical point of view each test presents a specific difficulty in terms of added mass effects.

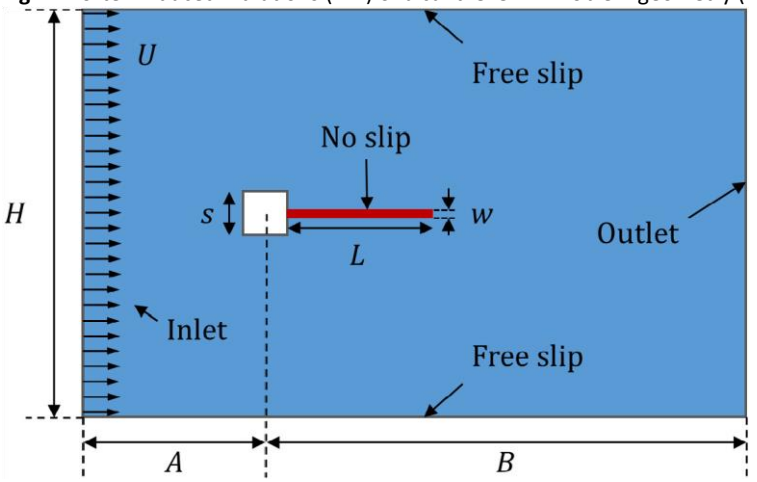
5.1. VORTEX-INDUCED VIBRATIONS (VIV) OF A CANTILEVER

First, the coupling of the PFEM and Metafor solvers through CUPyDO is tested against existing results from the literature. A classic FSI benchmark [2,66,73,74] is chosen as a validation test: the vortex-induced vibrations (VIV) of a cantilever. As depicted in Fig. 2, an elastic cantilever (red) is clamped to a rigid square base (white) and immersed in a steady flow (blue) of constant velocity, U, imposed at the inlet. The problem is designed in such a way that the frequency of vortex shedding induced by the presence of a bluff body (i.e. the square) is the same as the one of the first mode of vibration of the cantilever. Thus, a strong coupling between the fluid and the solid is expected, involving large displacements of the structure, which makes this a challenging problem. This problem has already been solved by the authors in a previous paper [20], employing the SU2 solver [75] – based on the finite volumes method (FVM) – for the fluid and Metafor for the solid. In the present case, Metafor is also used for the solid but the fluid equations are solved using PFEM instead of FVM. Note that the PFEM has not been originally designed to treat this kind of problems, as its Lagrangian nature is not ideally suited to dealing with flows crossing the computational domain boundaries (i.e., inlet and outlet) like in this test case. The PFEM has thus been adapted for this problem: at the inlet a constant amount of particles is continuously injected at velocity U, while at the outlet a "do-nothing" condition [76–78] is employed and particles that exit the domain are eliminated from the computation.

The different material properties for the solid and the fluid are listed in Table 1, while the FSI parameters are reported in Table 2. In this case a BGS algorithm without relaxation is sufficient to ensure the convergence of the coupled problem.

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

Fig. 2. Vortex-induced vibrations (VIV) of a cantilever — Problem geometry (initial configuration).



Some snapshots of the response of the system at different time steps are reported in Fig. 3 and compared to the results obtained using Metafor+SU2 [20]. The moving PFEM particles can be seen in the figures referring to the case Metafor+PFEM. The results are qualitatively in good agreement. Some differences can be observed especially far from the cantilever. In particular, the Metafor+PFEM solution seems to capture more details of the flow. This is due to the fact that the PFEM employs a discretization which is almost uniform over the whole domain while SU2 uses smaller cells close to the cantilever and larger cells far away from it, which prevents SU2 from capturing small details in those regions of the domain. Fig. 4 reports the time evolution of the vertical displacement of the tip of the cantilever for the case Metafor+PFEM and Metafor+SU2, respectively. Some differences can be observed especially in the initial transitory phase. This is not surprising since this case is characterized by a physical instability, thus the path leading to the limit cycle can be greatly affected by small numerical differences. To have a more significant and quantitative comparison, Table 3 compares the maximum value of the displacement of the cantilever tip and the vibration frequency of the cantilever obtained in this work with the results previously registered by the authors using SU2, and with those presented in other works from the literature, employing various combinations of numerical methods for the solid and the fluid. Although some dispersion exists among the different authors, the results obtained with Metafor+PFEM can be considered in good agreement with those from the literature.

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

Table 1. Vortex-induced vibrations (VIV) of a cantilever — Solid and fluid properties.

Geometry	Distance of the square center from inlet [m]	A	5.5×10^{-2}
	Distance of the square center from outlet [m]	B	14×10^{-2}
	Channel width [m]	H	12×10^{-2}
	Cantilever length [m]	L	4×10^{-2}
	Cantilever thickness [m]	w	0.06×10^{-2}
	Square side [m]	s	1×10^{-2}
	Inlet velocity [m/s]	U	0.513
Solid	Density [kg/m^3]	ρ_s	100
	Young's modulus [Pa]	E	2.5×10^5
	Poisson's ratio [–]	ν_s	0.35
	No. of EAS Elements		54×2
Fluid	Density [kg/m^3]	ρ_f	1.18 1.54 ×
	Kinematic viscosity [m^2/s]	ν_f	10^{-5}
	No. of particles		63678

Table 2. Vortex-induced vibrations (VIV) of a cantilever — FSI parameters.

Algorithm	BGS
Relaxation	None
Δt_{FSI} [s]	7.5×10^{-4}
ϵ_{FSI} [m]	1×10^{-6}

Table 3. Vortex-induced vibrations (VIV) of a cantilever — Results for the cantilever oscillation amplitude and frequency.

Author	Solid	Fluid	Max. tip displ. [mm]	Δ_{displ} [%]	Freq. [Hz]	Δ_{freq} [%]
CUPyDO: Metafor+PFEM	FEM	PFEM	1.15	–	3.25	–
CUPyDO: Metafor+SU2 [20]	FEM	FVM	1.07	−6.9	3.14	−3.4
Wood et al. [66]	FEM	FVM	1.15	0.0	2.94	−9.5
Olivier et al. [73]	FVM	FVM	0.95	−17.4	3.17	−2.4
Dettmer and Peric [´ 74]	FEM	FEM	1.25	+8.7	3.03	−6.7
Habchi et al. [2]	FVM	FVM	1.02	−11.3	3.25	0.0

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

Fig. 3. Vortex-induced vibrations (VIV) of a cantilever — Contour of the velocity magnitude at three instants of an oscillation cycle: Metafor+PFEM vs Metafor+SU2.

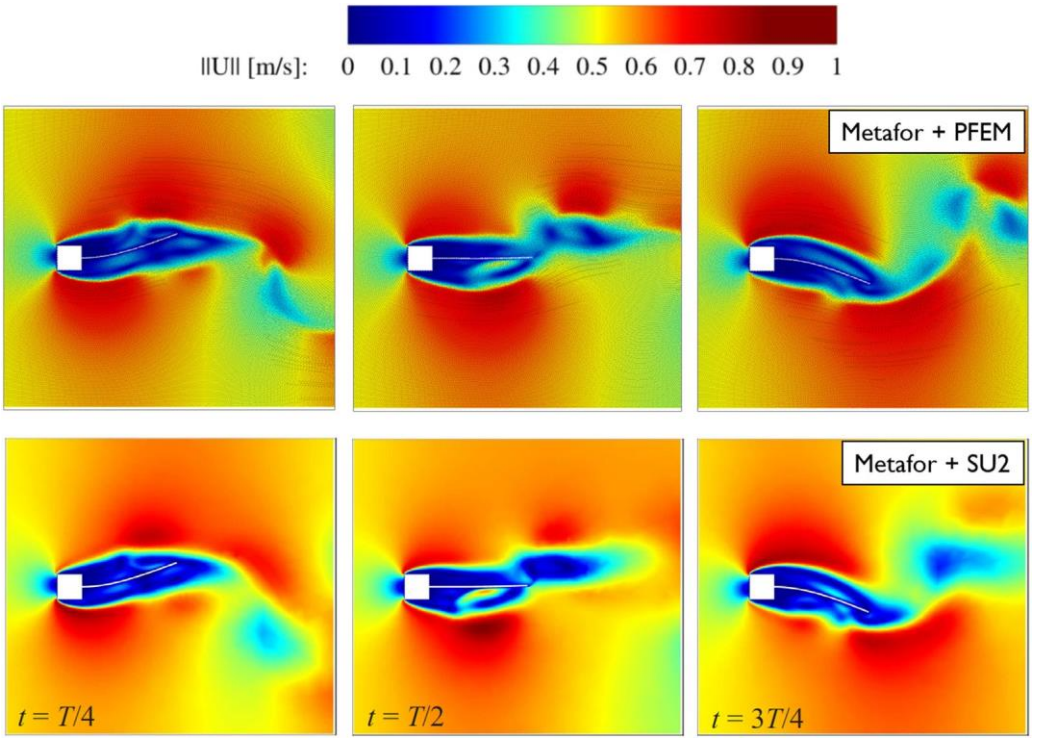
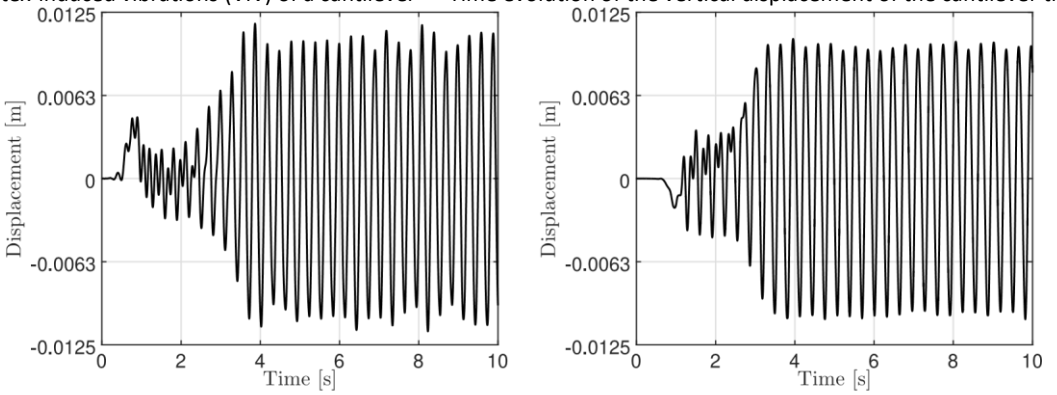


Fig. 4. Vortex-induced vibrations (VIV) of a cantilever — Time evolution of the vertical displacement of the cantilever tip.



(a) Metafor + PFEM (b) Metafor + SU2

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

5.2. DAM BREAK AGAINST AN ELASTIC–PLASTIC OBSTACLE

A further validation example considers a very simplified model of the breaking of a dam against a deformable obstacle. This benchmark was initially proposed by Walhorn et al. [79] who used a monolithic consistent space–time finite element formulation. The geometry of the problem is depicted in Fig. 5. A column of fluid is initially held in a reservoir by a rigid wall. At the beginning of the simulation the wall is instantaneously removed and the fluid is let free to flow. A deformable obstacle is clamped in the middle of the reservoir such that it is hit by the collapsing column of fluid. The fluid is water, with properties listed in Table 4. As for the solid, in the first version of the benchmark the obstacle was modeled with pure (large strain) elastic behavior. This version of the problem is usually employed as validation benchmark in the literature [1,16,18,19,80–82]. The benchmark has been then extended by Zhu and Scott [18] using an elastic–plastic material model with linear isotropic hardening, as shown in Fig. 6. Solid material properties are summarized in Table 4 as well, both for the elastic and plastic regimes. The FSI coupling is performed using a BGS implicit coupling with Aitken relaxation. The parameters employed for the coupling are listed in Table 5.

Table 4. Dam break against an elastic–plastic obstacle — Solid and fluid properties.

Geometry	Water column width [m]	L	0.146
	Obstacle width [m]	w	0.012
	Obstacle height [m]	h	0.080
	Gravity [m/s^2]	g	9.81
Solid	Density [kg/m^3]	ρ_s	2500
	Young's modulus [Pa]	E	1×10^6 0
	Poisson's ratio [–]	ν_s	5×10^4
	Yield stress [Pa]	σ_y	
	Plastic modulus [Pa]	E_p	2×10^4
	No. of SRI Elements		4 × 31 (reference), 10 × 64 (fine)
Fluid	Density [kg/m^3]	ρ_f	1000
	Dynamic viscosity [Pa · s]	μ_f	0.001
	No. of particles		2987 (reference), 11240 (fine)

Table 5. Dam break against an elastic–plastic obstacle — FSI parameters.

Algorithm	BGS
Relaxation	Aitken
ω_{max}	0.5
Δt_{FSI} [s]	1×10^{-3}
ϵ_{FSI} [m]	1×10^{-6}

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

Fig. 5. Dam break against an elastic–plastic obstacle — Problem geometry (initial configuration).

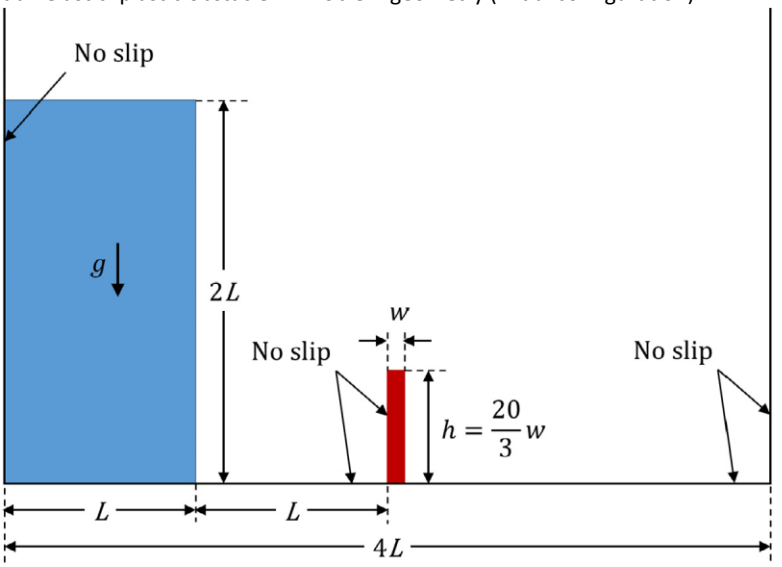
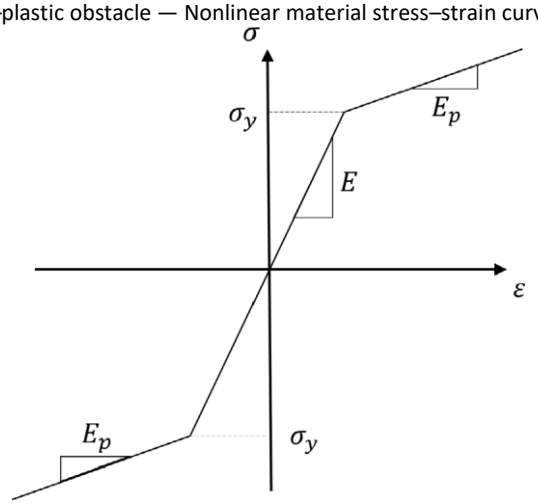


Fig. 6. Dam break against an elastic–plastic obstacle — Nonlinear material stress–strain curve [18].

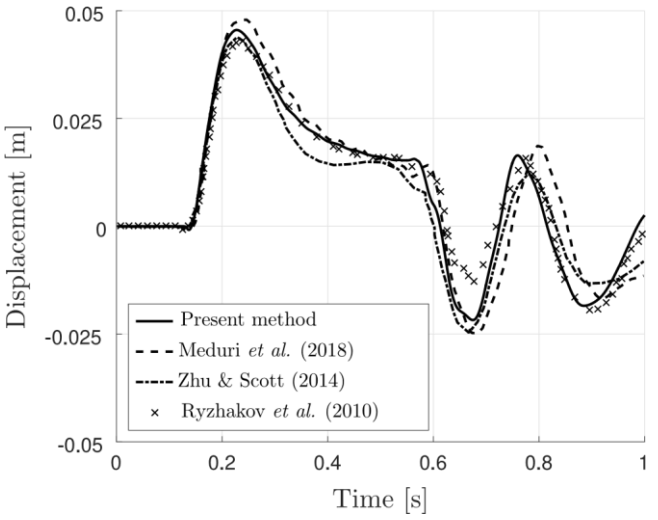


First the results obtained with the present approach are validated for the case of the dam break against an elastic obstacle. The time evolution of the horizontal displacement of the left upper corner of the obstacle is reported in Fig. 7 and compared with the results obtained by Meduri et al. [19] (explicit FEM/PFEM coupling with Lagrange multipliers), Zhu and Scott [18] (monolithic FEM/PFEM coupling with fractional step), and Ryzhakov et al. (monolithic PFEM). Despite some dispersion in the reference results,

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

the amplitude of the initial peak and the overall time evolution of the response compare well with the literature.

Fig. 7. Dam break against an elastic–plastic obstacle — Validation: results for a purely elastic obstacle. Horizontal displacement of the left upper corner of the obstacle as a function of time.



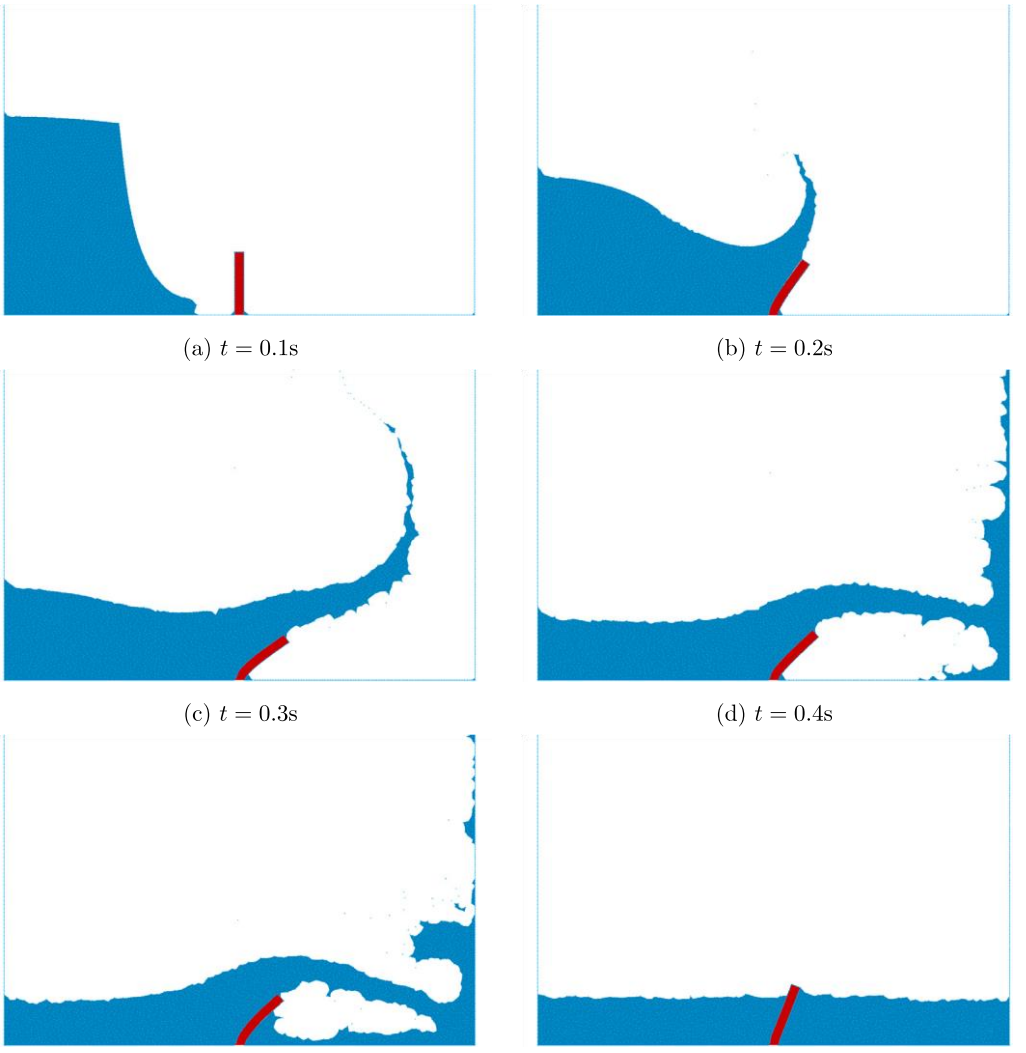
The same simulation is then run again using an elastic–plastic constitutive model for the obstacle. This is one of the main advantages of a partitioned coupling: any constitutive model can be used for the solid or the fluid, only depending on the capabilities of the solvers employed, without any consequence on the FSI coupling itself. Snapshots of the solution obtained for the elastic–plastic obstacle at different time steps are collected in Fig. 8.

The time evolution of the horizontal displacement of the tip of the obstacle for this case is shown in Fig. 9. Again, the agreement with the results obtained by Zhu and Scott [18] is very good in the first phase of the interaction between the water and the solid, up to around 0.75 s. After this time, the two approaches give quite different results. The final value of the displacement predicted by the two approaches is also different. Zhu and Scott predict a final displacement of around 0.02 m, while the present approach estimates its value around 0.03 m. This difference can be mainly explained by the fact that Zhu and Scott employ a corotational mesh of beam elements to model the obstacle. The thickness of the obstacle is thus not taken into account on the fluid side, introducing a non-negligible difference in the geometry of the problem, especially once the water has passed the obstacle. To verify the results obtained with the present approach, the same simulation has been performed using a much finer mesh (details can be found in Table 4). Fig. 10 shows the comparison between the responses obtained with the two discretizations.

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

The results obtained with the finer mesh confirm the ones obtained with the reference mesh, indicating that the spatial discretization is fine enough to capture the relevant physics.

Fig. 8. Dam break against an elastic–plastic obstacle — Snapshots of the simulation at different times.



5.3. FLUID COLUMN ON AN ELASTIC COLUMN

This example represents a first test to investigate the performance of the proposed approach for problems characterized by strong added-mass effects, as introduced in Section 4. This benchmark was proposed by

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

Idelsohn et al. [30] to test a modified fractional-step method that alleviates added-mass effects (cf. end of Section 4). The problem is depicted in Fig. 11 and consists of a column of inviscid incompressible fluid that lies on an elastic solid column. The two vertical sides of the whole domain are blocked in the horizontal direction, the upper fluid side is a free surface and the bottom solid side is clamped. A vertical body force is applied to both the solid and the fluid domains. The undeformed initial configuration and the absence of any dissipation (i.e. inviscid fluid + elastic solid + no friction) should lead to periodic oscillations forced by gravity. The geometry and the different material parameters are listed in Table 6.

Fig. 9. Dam break against an elastic–plastic obstacle — Time evolution of the horizontal displacement of the upper left corner of the elastic–plastic obstacle.

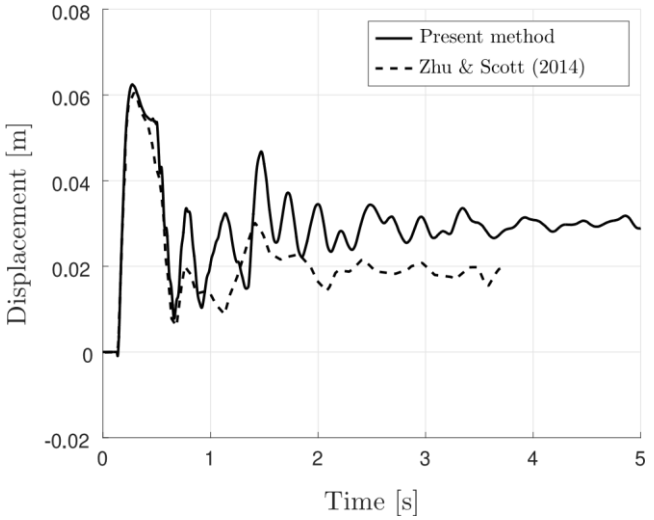
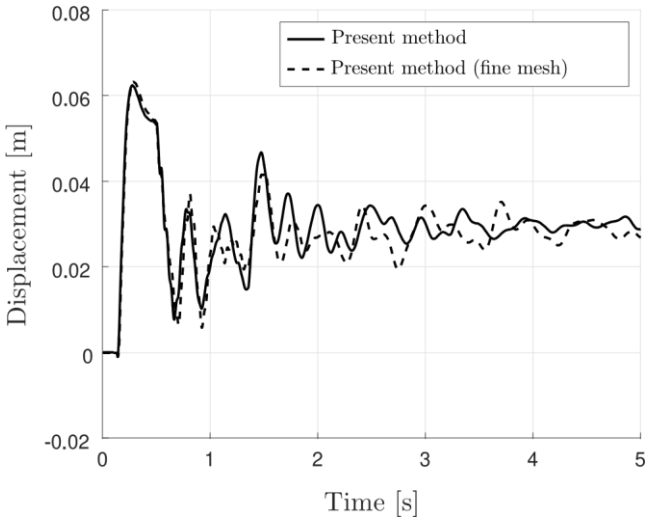


Fig. 10. Dam break against an elastic–plastic obstacle — Time evolution of the horizontal displacement of the obstacle tip. Reference mesh vs fine mesh.



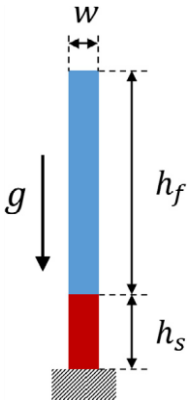
Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

The problem has been designed to exacerbate added-mass effects. The fluid and solid densities are very similar, the solid Young's modulus is very low (2.3×10^5 Pa corresponds to a very soft material, like muscles for instance), and there is no physical dissipation. For these reasons the same benchmark has also been employed by Zhu and Scott [18] to assess the performance of an enhanced fractional step method for FSI problems and by Meduri et al. [83] who employ a fully explicit FSI coupling through Lagrange multipliers.

Table 6. Fluid column on an elastic column — Solid and fluid properties.

Geometry	Column width [m]	w	0.05
	Solid column height [m]	h_s	0.25
	Water column height [m]	h_f	0.75
	Gravity [m/s^2]	g	10
Solid	Density [kg/m^3]	ρ_s	1500
	Young's modulus [Pa]	E	2.3×10^5
	Poisson's ratio [–]	ν	0.4
	No. of Linear Triangles		40
Fluid	Density [kg/m^3]	ρ_f	1000
	Dynamic viscosity [Pa · s]	μ_f	0
	No. of particles		93

Fig. 11. Fluid column on an elastic column — Problem geometry (initial configuration).



In this work the partitioned approaches described in Section 4, in particular the IQN-ILS technique, are tested against this benchmark which, despite its apparent simplicity, is not easily solved using traditional coupling techniques, either implicit or explicit, as discussed by Idelsohn et al. [30]. The FSI parameters used in the simulation are collected in Table 7. In this case, the BGS method is also employed without

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

relaxation and with (very low) static relaxation, to demonstrate the difficulties encountered by traditional partitioned coupling techniques to solve this problem. In the following, *BGS* implies BGS coupling with Aitken relaxation. When no relaxation or static relaxation is employed, it will be explicitly indicated.

The time evolution of the vertical displacement of the upper side of the solid column obtained with the present approach using a BGS coupling is shown in Fig. 12 and compared to the results obtained by Meduri et al. [83] (explicit FEM/PFEM coupling with Lagrange multipliers), Zhu and Scott [18] (Monolithic FEM/PFEM coupling with fractional step), and Idelsohn et al. [30] (Monolithic PFEM with fractional step). The solution obtained by CUPyDO is in very good agreement with the results of Meduri et al. and Idelsohn et al., while the response obtained by Zhu and Scott is quite different from all the others, for reasons unknown to the authors.

Table 7. Fluid column on an elastic column — FSI parameters.

Algorithm	BGS	IQN-ILS
Relaxation	None/Static/Aitken	Static
ω_{max}	$-/1 \times 10^{-2}/0.5$	0.5
Filtering	–	Haelterman
Filtering tolerance	–	1×10^{-1}
Δt_{FSI} [s]	1×10^{-5}	1×10^{-5}
ϵ_{FSI} [m]	1×10^{-7}	1×10^{-7}

Fig. 12. Fluid column on an elastic column — Time evolution of the vertical displacement of the solid–fluid interface.

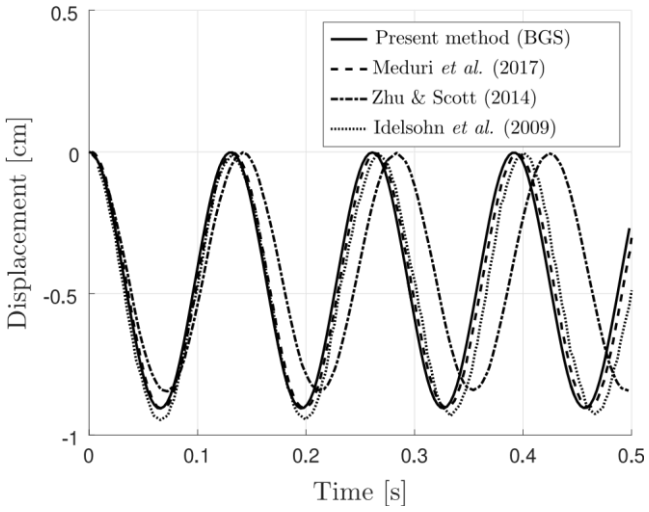


Fig. 13 compares the response obtained using BGS with the one obtained with IQN-ILS coupling, without any information from the previous time steps, and exploiting the information from 10 previous time steps.

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

The results are exactly the same for all three methods. Table 8, on the other hand, shows the mean number of FSI iterations per time step for each of the tested coupling strategies. First, if BGS with no relaxation is used, divergence of the solution is observed after two time steps only. Things are slightly better with static relaxation but the coupling algorithm still fails to converge after 21 time steps. Conversely, BGS with Aitken relaxation and IQN-ILS show very good performance with respect to added-mass effects for this problem, as less than 3 iterations on average are necessary to converge. It can also be noticed that in this case no difference exists between BGS and IQN-ILS results, which can be explained by the relatively small number of iterations required to solve this particular problem. Indeed, when the IQN-ILS method does not use information from previous time steps, at least two FSI iterations are needed to build a first approximation of the inverse tangent matrix. As expected, things are different for the IQN-ILS(10), which needs, on average, one iteration less than the other methods to converge.

Fig. 13. Fluid column on an elastic column — Time evolution of the vertical displacement of the solid–fluid interface. BGS vs IQN-ILS.

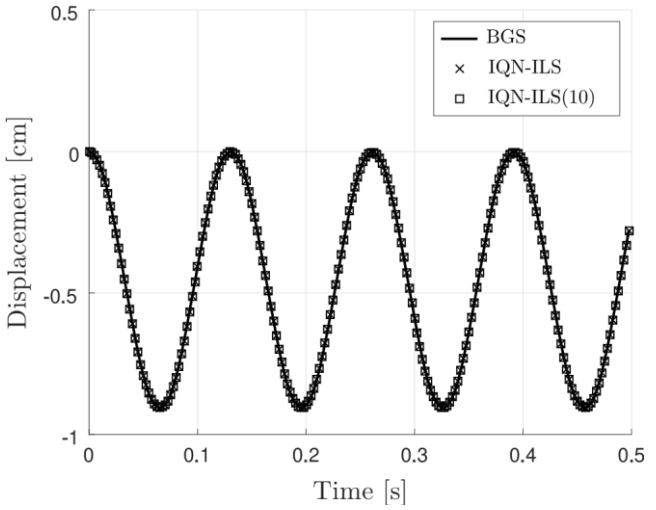


Table 8. Fluid column on an elastic column — Mean No. of FSI iterations.

Algorithm	Mean No. of iterations [–]
BGS — no relaxation	Does not converge (at time step 2)
BGS — static relaxation	Does not converge (at time step 21)
BGS — Aitken relaxation	2.92
IQN-ILS	2.92
IQN-ILS(10)	1.96

Idelsohn et al. [30] proposed a series of variations of the same problem in order to further analyze the performance of their fractional step approach in terms of added-mass effects. In particular, they investigated different combinations of stiffnesses, density ratios, and time step sizes. Tables 9–11 compare the average number of iterations over 1000 time steps obtained with BGS and IQN-ILS(10) and

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

the number of nonlinear iterations of the fractional step scheme of Idelsohn et al. [30] and of the enhanced fractional step scheme of Zhu and Scott [18]. Note that the number of iterations reported by the last two references is integer with no additional information, so that it has been assumed to correspond to a measure obtained at one given time step.

Table 9. Fluid column on an elastic column — FSI (columns 2–3) and fractional-step (columns 4–5) iterations for different density ratios.

ρ_s/ρ_f	BGS-Aitken	IQN-ILS(10)	Fractional step [18]	Fractional step [30]
10	1.91	1.88	3	20
7	2.53	1.90	3	20
6	1.96	1.92	3	20
5	2.78	1.91	3	20
3	2.48	1.94	3	19
1	4.18	1.99	3	18

$E = 2 \times 10^{11}$ Pa; $\nu = 0.3$; $\Delta t = 10^{-5}$ s.

Table 10. Fluid column on an elastic column — FSI (columns 2–3) and fractional-step (columns 4–5) iterations for different stiffnesses.

E [Pa]; Δt [s]	BGS-Aitken	IQN-ILS(10)	Fractional step [18]	Fractional step [30]
2×10^{13}; 0.2×10^{-5}	1.95	1.94	3	10
2×10^{12}; 0.5×10^{-5}	2.19	1.93	3	14
2×10^{11}; 1.0×10^{-5}	4.18	1.99	3	18
2×10^{8}; 1.0×10^{-4}	4.41	2.18	4	40
2×10^{7}; 1.0×10^{-3}	4.66	2.02	5	36
2×10^{6}; 1.0×10^{-3}	4.62	2.42	5	40
2×10^{6}; 1.0×10^{-2}	3.93	2.06	11	34
2×10^{5}; 1.0×10^{-2}	6.32	3.34	8	36

Table 11. Fluid column on an elastic column — FSI (columns 2–3) and fractional-step (columns 4–5) iterations for different time step sizes.

Δt [s]	BGS-Aitken	IQN-ILS(10)	Fractional step [18]	Fractional step [30]
2.0×10^{-5}	1.71	1.86	3	23
1.0×10^{-5}	2.53	1.90	3	20
0.75×10^{-5}	2.20	1.88	3	18
0.5×10^{-5}	2.60	1.84	3	16
0.25×10^{-5}	2.68	1.83	3	11

$\rho_s/\rho_f = 7$; $E = 2 \times 10^{11}$ Pa; $\nu = 0.3$.

The results obtained show the very good performance of the partitioned coupling implemented in CUPyDO, and in particular of the IQN-ILS technique that, for all the tests performed, needed at most around 3 FSI iterations per time step on average to converge. However, it is important to stress out that

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

this comparison, although indicative of general trends and useful for future studies, is not quantitatively meaningful as it relies on different measures of convergence between different approaches.

Table 12. Dam break through an elastic gate — Solid and fluid properties.

Geometry	Water column width [m]	A	0.100
	Water column height [m]	H	0.140
	Gate width [m]	s	0.005
	Gate height [m]	L	0.079
	Gravity [m/s^2]	g	9.81
Solid	Density [kg/m^3]	ρ_s	1100
	Material		Mooney–Rivlin
	No. of Linear Quad Elements		4 × 39
Fluid	Density [kg/m^3]	ρ_f	1000
	Dynamic viscosity [Pa · s]	μ_f	0.001
	No. of particles		5178

5.4. DAM BREAK THROUGH AN ELASTIC GATE

Similarly to the example presented in Section 5.2, the collapse of a column of water under the action of gravity is considered in this example. This time though, an elastic gate initially retains the column of water. The geometry of the problem is depicted in Fig. 14. This problem has been initially proposed by Antoci et al. [84], who investigated it both experimentally and numerically, using a fluid/solid SPH model. The geometrical and material parameters used in the simulation are taken from the experimental setup of Antoci et al. [84] and are listed in Table 12. The fluid is water. The solid is rubber and the material behavior is modeled as a hyperelastic solid of the Mooney–Rivlin family. The constitutive law is defined by fitting the experimental data – obtained through a tension test – provided by Yang et al. [85] (taken from Antoci [86]), and reported in Fig. 15. Since no further information is available, the authors assume that the experimental measures are expressed in true strains and true stresses. The results given by assuming linear elastic behavior with an estimated Young's modulus of 10MPa at the origin (as proposed by Antoci et al. [84]) are also added in the figure, to show that the hypothesis of linear elastic behavior becomes quickly very inaccurate, as soon as the solid undergoes even moderately large deformations. This fact has already been discussed in the literature, for instance by Meduri et al. [19], Yang et al. [85], and Li et al. [1]. Finally, the contact between the gate and the bottom wall has not been taken into account in the numerical model. The FSI parameters employed in the simulation are reported in Table 13. Again, both BGS and IQN-ILS techniques are tested. Notice that in this case the solid density is very close to the one of the fluid ($\rho_s/\rho_f = 1.1$) and that the gate is a slender highly flexible structure. Thus, as discussed in Section

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

4, this is a problem where added-mass effects can be critical, if no precaution is taken with respect to the partitioned coupling strategy.

Fig. 14. Dam break through an elastic gate — Problem geometry (initial configuration).

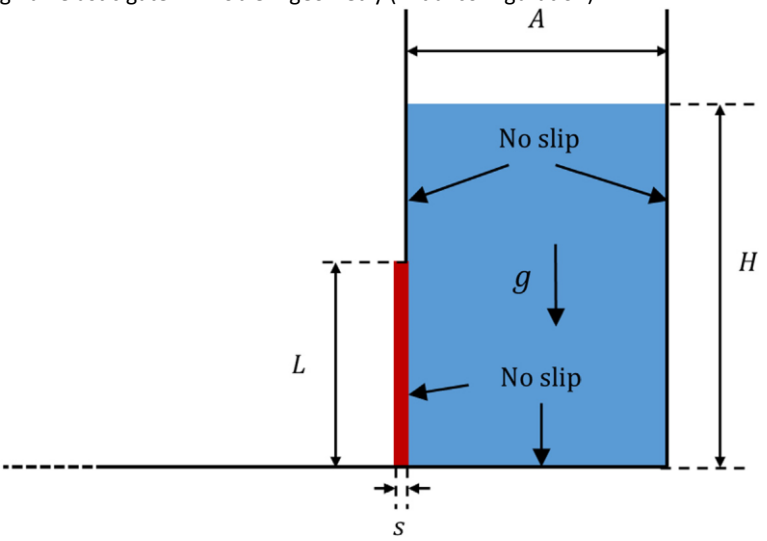
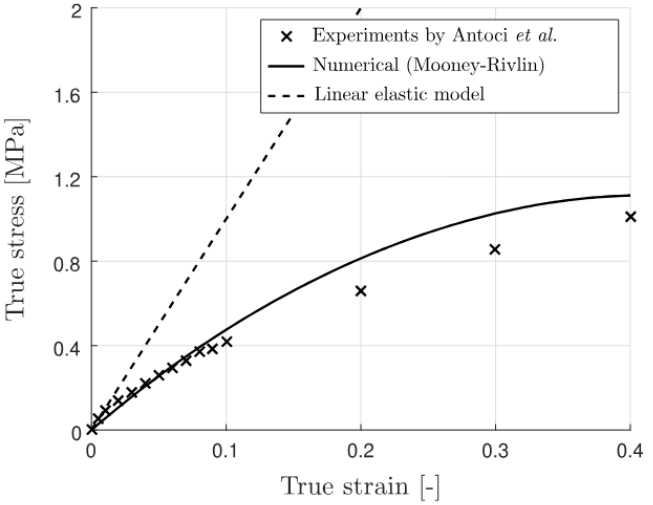


Fig. 15. Dam break through an elastic gate — Rubber traction test [86].



Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

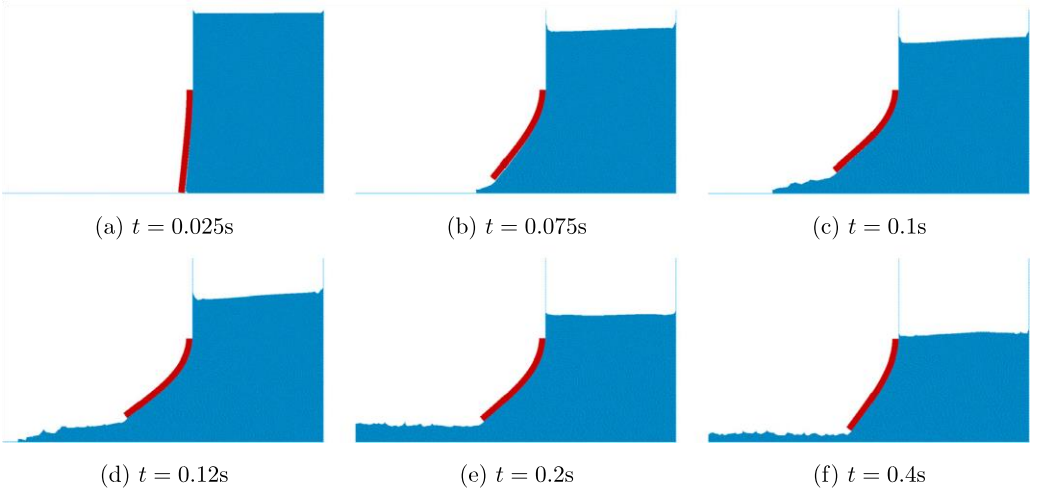
Table 13. Dam break through an elastic gate — FSI parameters.

Algorithm	BGS	IQN-ILS
Relaxation	Aitken	Static
ω_{max}	0.5	0.5
Filtering	–	Degroote/Haelterman
Filtering tolerance — Degroote	–	1×10^{-3}
Filtering tolerance — Haelterman	–	1×10^{-1}
No. of previous steps retained	–	0/1/5/10/20/30
Δt_{FSI} [s]	1×10^{-3}	1×10^{-3}
ε_{FSI} [m]	1×10^{-6}	1×10^{-6}

Moreover, this example will be exploited further to investigate some numerical parameters employed by the IQNILS strategy. In particular, the influence of the number of previous time steps used in the approximation of the inverse tangent matrix and of the different filtering strategies mentioned in Section 4 will be analyzed.

Fig. 16 shows some time instants of the simulation run with BGS coupling and Aitken relaxation. Initially, the gate is held fixed and the column of water is thus in static equilibrium. Suddenly, the gate is released and it starts deforming under the action of the pressure exerted by the water submitted to its own weight (t = 0.025 s). At around t = 0.075 s the gate is fully open and the water can flow through the gate. The gate continues deforming under the action of water pressure and of its own inertia, up to about t = 0.12 s. After this point the gate starts closing again because the level of water has decreased and, with it, the pressure applied on the gate. The flow rate through the gate is thus reduced as shown in Figs. 16(e) and 16(f).

Fig. 16. Dam break through an elastic gate — Snapshots of the simulation at different times. Results obtained using BGS with Aitken relaxation.



Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

The same conclusions can be drawn by looking at the time evolution of the horizontal and vertical displacements of the lower left tip of the elastic gate, reported in Fig. 17. The results obtained in the present work are compared to the experimental and numerical results of Antoci et al. [84] showing very good agreement. The difference between the numerical and experimental results in the prediction of the maximum displacements can be explained by looking at the photos of the experiment in Antoci et al. [84]. Some water flowing through the sides of the gate, which is not perfectly sealed, can be observed at the beginning of the experiment, thus inducing a lower pressure on the gate than the one computed numerically where no side flow is present.

Fig. 17. Dam break through an elastic gate — Time evolution of the displacements of the gate tip. Experimental results taken from Antoci et al. [84].

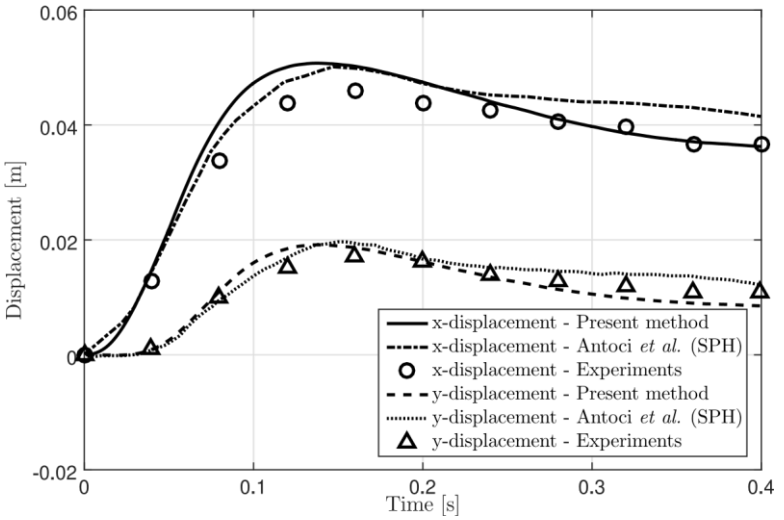


Table 14 lists the mean number of FSI iterations per time step, and the corresponding CPU times,[1] obtained for this problem using BGS coupling with Aitken relaxation and different versions of IQN-ILS (the same quantities, normalized with respect to those corresponding to BGS with Aitken relaxation, are reported between parentheses to ease the interpretation of the results). Table 14 shows that the main cost in a partitioned FSI coupling is associated to the mean number of FSI iterations per time step, or, in other words, to how many times the solid and (usually more importantly) the fluid solvers are run within each time step, and that the complexity added by the IQN-ILS method is negligible compared to the gain obtained in terms of number of FSI iterations. For instance, IQN-ILS(5) with *Haelterman* filtering leads to a 29% gain in terms of number of FSI iterations and to a 27% gain in terms of CPU time, compared to BGS with Aitken relaxation. Similar conclusions can be drawn for all the other cases.

[1] Simulations were run on a 3.20 GHz Intel Core i5-4570 processor.

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

For the IQN-ILS technique, the two different filtering strategies described in Section 4 are tested. As it can be observed, the use of *Haelterman* filtering [70] globally enhances the performance of the IQN-ILS strategy, ensuring faster convergence (i.e. lower number of FSI iterations per time step) and lower computational costs. Additionally, the use of different numbers of previous time steps employed in the approximation of the inverse Jacobian is analyzed. In general, convergence should be accelerated by taking into account information coming from an increasing number of previous time steps. This is true when using *Haelterman* filtering, while *Degroote* filtering does not show a monotonic trend, which makes this approach more difficult to use in practice, since it is impossible to know a priori the optimal number of time steps to retain for a given filtering tolerance. In this case, for instance, the best performance with *Degroote* filtering is obtained for IQN-ILS(1). As a last remark, the use of an extremely large number of previous time steps should be avoided, even in the case of *Haelterman* filtering. This is demonstrated by Table 14 which shows that, independently of the filtering technique, the CPU times decrease to a lower extent than the number of FSI iterations when a larger number of previous time steps is employed. In some cases, a lower number of FSI iterations leads even to a higher CPU time. This is due to the fact that the use of a larger number of previous time steps necessarily leads to the construction of larger V and W matrices. More time is thus spent in the factorization (and related filtering operations) of the V matrix, which in part counterbalances the gain obtained in terms of FSI iterations.

Table 14. Dam break through an elastic gate — Mean No. of FSI iterations per time step and CPU times. Between parentheses, the results normalized with respect to the BGS solution.

Algorithm	Filtering	Mean No. of iterations (normalized)	CPU [s] (normalized)
BGS	–	5.43 (1)	3460.29 (1)
IQN-ILS(0)	Degroote	4.83 (0.889)	3004.11 (0.868)
IQN-ILS(1)	Degroote	4.03 (0.742)	2617.74 (0.756)
IQN-ILS(5)	Degroote	4.21 (0.775)	2649.91 (0.767)
IQN-ILS(10)	Degroote	4.22 (0.777)	2705.89 (0.782)
IQN-ILS(20)	Degroote	4.15 (0.764)	2691.50 (0.778)
IQN-ILS(30)	Degroote	4.11 (0.757)	2660.57 (0.768)
IQN-ILS(0)	Haelterman	4.87 (0.897)	2955.99 (0.854)
IQN-ILS(1)	Haelterman	3.94 (0.726)	2543.89 (0.735)
IQN-ILS(5)	Haelterman	3.88 (0.715)	2522.15 (0.729)
IQN-ILS(10)	Haelterman	3.86 (0.711)	2563.58 (0.741)
IQN-ILS(20)	Haelterman	3.85 (0.709)	2605.65 (0.753)
IQN-ILS(30)	Haelterman	3.82 (0.703)	2546.18 (0.736)

5.5. CYLINDER POP-OFF

In this example CUPyDO is tested against a problem involving a very light solid moving into a heavier fluid, i.e. the "pop-off" of a cylinder, initially immersed in an open reservoir. The geometry of the problem is depicted in Fig. 18(a). The cylinder is initially placed at a given depth, δ, and then suddenly released and

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

let free to move. The upper surface of the reservoir is a free surface. Gravity is applied to the solid–fluid system, thus buoyancy is the force driving the motion. If the cylinder is lighter than the fluid (i.e. solid-to-fluid density ratio, $\rho_s/\rho_f < 1$) it will move upwards, otherwise (i.e. $\rho_s/\rho_f > 1$) it will move downwards. In case of an upward motion, the cylinder can potentially cross the interface, possibly pop-off outside the fluid bulk, and then eventually fall back down. This is a very complex problem from two points of view: the strongly coupled interaction between the solid and the fluid when the cylinder moves inside the bulk, and the interaction between a moving solid and an evolving fluid interface that can be traversed by the solid itself.

Before analyzing the pop-off problem, the case of a falling cylinder ($\rho_s/\rho_f > 1$) in a *closed* reservoir is considered for validation. The problem is depicted in Fig. 18(b). The geometrical, material, and numerical parameters are listed in Table 15. The solid is modeled as a very stiff elastic material. In this case, since the fluid is viscous, the cylinder is expected to accelerate under the action of gravity until it reaches a terminal velocity. This problem has already been studied in the literature by Franci et al. [17], Hesch et al. [87], Gil et al. [88], and Wang and Liu [89], who found an asymptotic value for the velocity of about 0.037 m/s. The FSI coupling parameters employed in this example are summarized in Table 16.

Fig. 19 shows the evolution of the simulation at different time steps, while Fig. 20 shows the time evolution of the vertical velocity of the cylinder, measured at its center of mass. As expected, the cylinder reaches an asymptotic velocity. The value obtained in this work is 0.03725 m/s, which is thus in perfect agreement with the results found in the literature.

Table 15. Falling cylinder and pop-off — Solid and fluid properties.

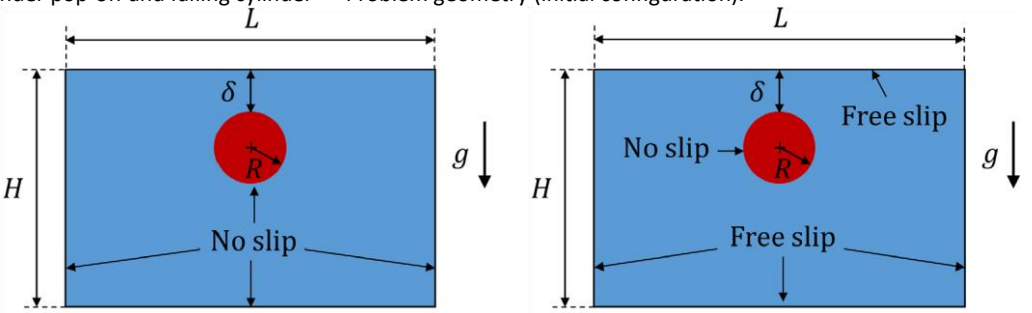
			Falling	Pop-off
Geometry	Reservoir width [m]	L	0.04	0.4
	Reservoir height [m]	H	0.075	0.2
	Cylinder diameter [m]	D	0.005	0.04
	Cylinder depth [m]	δ	0.005	0.04
	Gravity [m/s^2]	g	9.81	9.81
Solid	Density [kg/m^3]	ρ_s	1200	80
	Young's modulus [Pa]	E	1×10^9	1×10^8
	Poisson's ratio [–]	ν	0.45	0.4
	No. of Linear Triangles		86	5264
Fluid	Density [kg/m^3]	ρ_f	1000	1000
	Dynamic viscosity [Pa · s]	μ_f	0.1	0.001
	Surface tension coefficient [N/m]	γ	–	72.86×10^{-3}
	No. of particles		4911	56444

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

Table 16. Falling cylinder — FSI parameters.

Algorithm	BGS
Relaxation	Aitken
ω_{max}	0.5
Δt_{FSI} [s]	1×10^{-3}
ε_{FSI} [m]	1×10^{-6}

Fig. 18. Cylinder pop-off and falling cylinder — Problem geometry (initial configuration).



(a) Cylinder pop-off (b) Falling cylinder

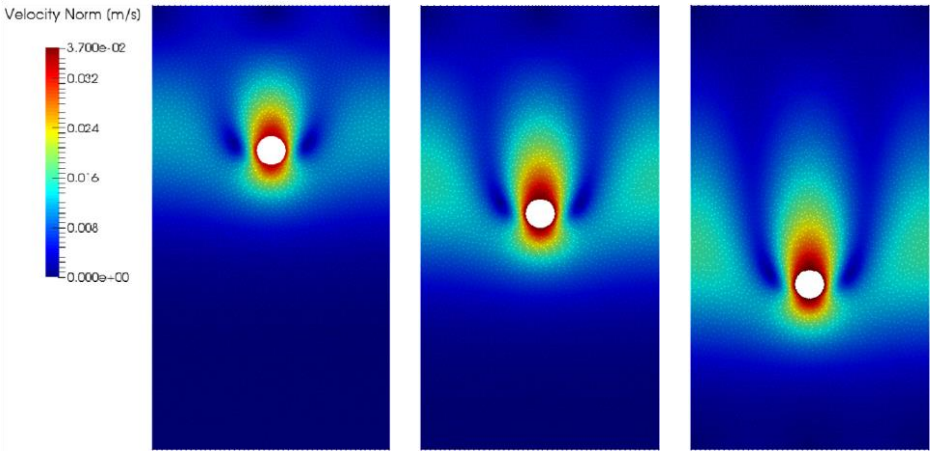
Table 17. Cylinder pop-off — FSI parameters.

Algorithm	BGS	IQN-ILS
Relaxation	Aitken	Static
ω_{max}	0.5	0.5
Filtering	–	Haelterman/Degroote
Filtering tolerance	–	1×10^{-1}
No. of previous steps retained	–	10/20/30
Δt_{FSI} [s]	1×10^{-3}	1×10^{-3}
ε_{FSI} [m]	1×10^{-5}	1×10^{-5}

The material and geometrical parameters defining the case of the cylinder pop-off are listed in Table 15. Surface tension effects are introduced at the free surface, assuming the presence of air all around the reservoir, as indicated by the surface tension coefficient reported in Table 15. Nonetheless, the air itself is not modeled. Again, the solid is modeled as a very stiff elastic material. Notice that the solid-to-fluid density ratio is extremely low ($\rho_s/\rho_f = 0.08$), corresponding approximately to a ping-pong ball in water. The FSI coupling parameters related to this example are summarized in Table 17.

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

Fig. 19. Falling cylinder — Contour of the velocity magnitude at three instants in time.



(a) $t = 0.375$s (b) $t = 0.675$s (c) $t = 1.0$s

Fig. 20. Falling Cylinder — Time evolution of the vertical velocity, measured at the center of mass of the cylinder.

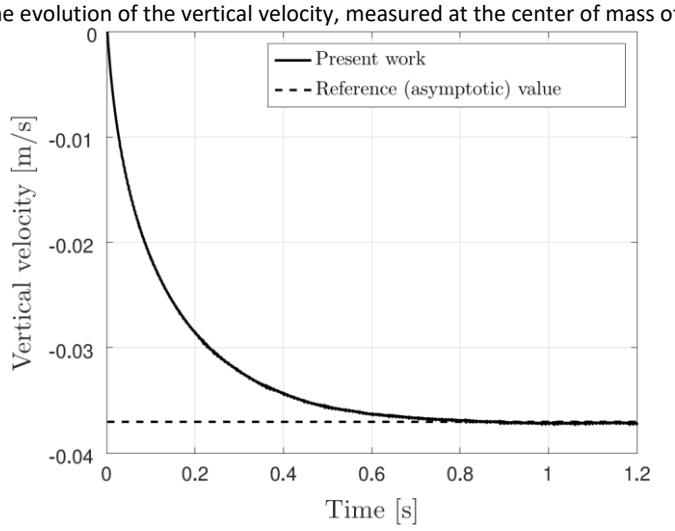


Fig. 21 shows some snapshots of the simulation. At around t = 0.1 s the cylinder reaches the initial position of the free surface, entraining some water at the same time. At around t = 0.15 s the cylinder has completely traversed the initial free surface and still entrains a film of fluid outside the bulk. The cylinder continues to rise outside the bulk until it reaches a maximum height at t = 0.25 s. At this moment, almost all the water has drained away from the cylinder. Then, the cylinder starts to fall back down (t = 0.30 s) until it impacts onto the water surface at around t = 0.35 s. From this moment on the cylinder floats on the water surface and its motion is mainly governed by surface waves until the end of the simulation (t =

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

1.0 s). Table 18 reports the values of the vertical velocity of the cylinder when it crosses the initial position of the fluid surface and the maximum height reached by the cylinder with respect to the initial position of the fluid surface. In both cases the top of the cylinder is considered. The maximum height reached by the top side of the cylinder normalized with respect to the cylinder diameter (h_{max}/D) is also included in the table for completeness. For the IQN-ILS case, simulations were run using 10, 20 and 30 previous time steps. A small dispersion in the results can be observed. Thus, average values are reported in the last row of Table 18. A maximum deviation of about 0.4% for the crossing velocity and 4% for the maximum height is registered. This can be explained by the fact that the FSI coupling always introduces an error, which is at most equal to the chosen tolerance ε_{FSI}, but which is never the same between the different coupling strategies. These small differences, in such a sensitive problem, are sufficient to lead to slightly different results.

Table 18. Cylinder pop-off — Cylinder crossing velocity and maximum height. $\overline{v}_y$: vertical velocity when the cylinder crosses the fluid surface. hmax: maximum height reached by the top side of the cylinder.

Algorithm	$\overline{v}_y$ [m/s]	h_{max} [m]	h_{max}/D [–]
BGS	0.721	0.0868	2.167
IQN-ILS(10)-Haelterman	0.727	0.0854	2.135
IQN-ILS(20)-Haelterman	0.724	0.0885	2.213
IQN-ILS(30)-Haelterman	0.724	0.0824	2.060
IQN-ILS(50)-Haelterman	0.727	0.0888	2.219
IQN-ILS(10)-Degroote	0.724	0.0829	2.073
IQN-ILS(20)-Degroote	0.727	0.0853	2.133
IQN-ILS(30)-Degroote	0.723	0.0823	2.058
IQN-ILS(50)-Degroote	0.724	0.0859	2.148
Average values	0.725	0.0854	2.134

Table 19. Cylinder pop-off — Mean No. of FSI iterations.

Algorithm	Mean No. of iterations [–]
BGS	3.18
IQN-ILS(10)-Haelterman	2.78
IQN-ILS(20)-Haelterman	2.69
IQN-ILS(30)-Haelterman	2.70
IQN-ILS(50)-Haelterman	2.69
IQN-ILS(10)-Degroote	2.69
IQN-ILS(20)-Degroote	2.70
IQN-ILS(30)-Degroote	2.71
IQN-ILS(50)-Degroote	2.70

The different phases of the simulation also clearly appear in the time evolution of the number of FSI iterations, as shown in Fig. 22 for the BGS and IQN-ILS(30) couplings. As it could be expected, in both cases one can notice a higher number of iterations around 0.1 s and 0.35 s, i.e. when the cylinder exits the bulk

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

and when the cylinder impacts onto the water surface after falling down, respectively. For the BGS coupling the number of iterations also raises around 0.45 s and 0.625 s, and for the IQN-ILS(30) coupling around 0.83 s. In both cases, this is mainly due to the interaction between the cylinder and the water surface waves in both cases. Overall, the IQN-ILS strategy is again more efficient than BGS, as confirmed by the mean number of FSI iterations reported in Table 19, although IQN-ILS(30) needs on average 0.01 iterations more than IQN-ILS(20) to converge. This difference can be considered negligible and does not contradict the conclusions drawn in example 5.4 about the monotonic trend obtained with *Haelterman* filtering in terms of FSI iterations reduction when a larger number of previous steps is employed. This merely indicates that in this case a value around 2.70 represents the minimum possible number of FSI iterations for the IQN-ILS technique and that the use of 20 previous time steps is already sufficient to obtain it.

5.6. FILLING OF AN ELASTIC CONTAINER

In this last example CUPyDO is tested in the case of a challenging FSI problem. The case has been initially simulated by Franci et al. [17] using a unified PFEM solid/fluid formulation, and it is inspired by a former problem presented by Cremonesi et al. [42]. More recently the same problem has also been used as a benchmark by Meduri et al. [19] using an explicit FEM/PFEM coupling with Lagrange multipliers.

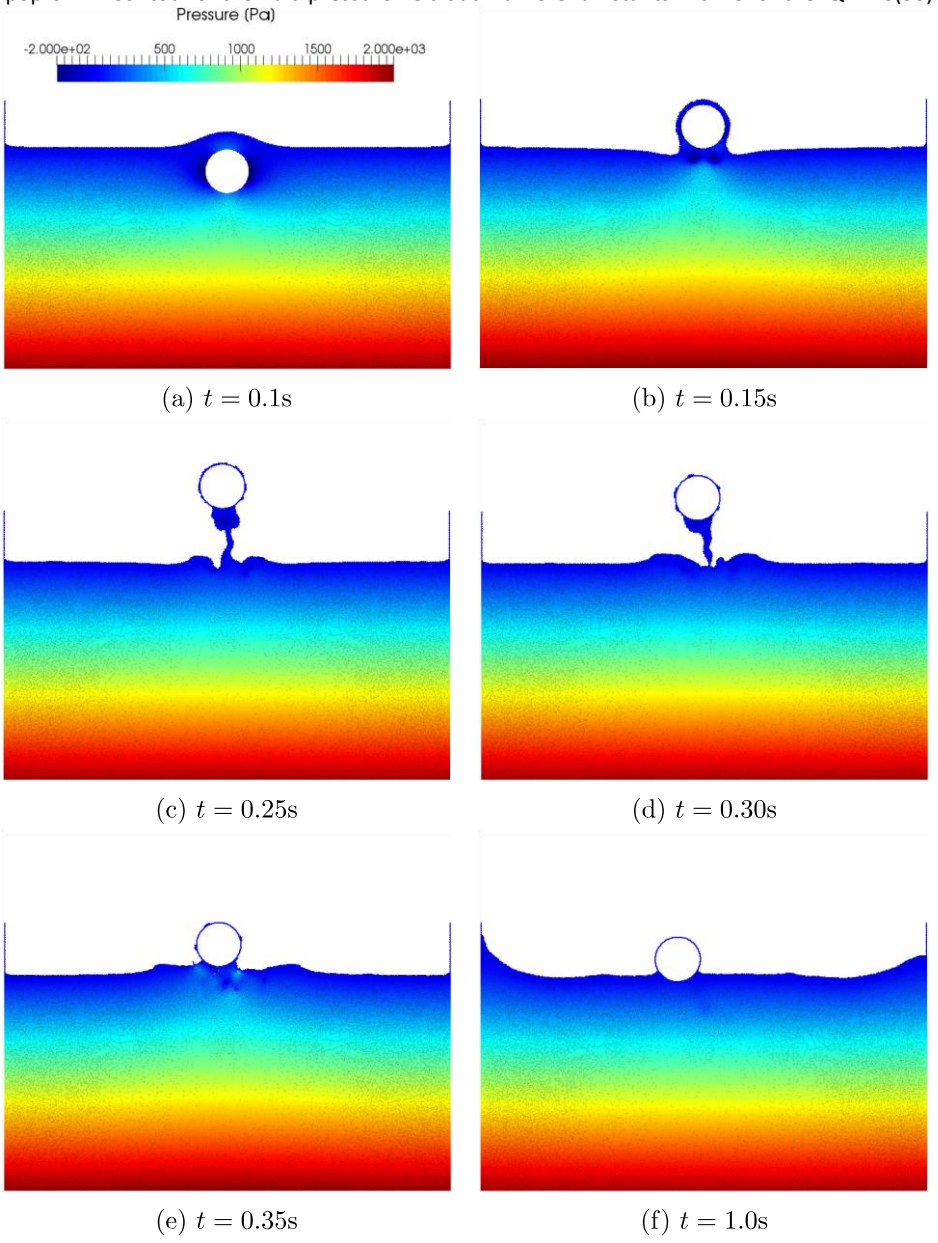
The geometry of the problem is depicted in Fig. 23. A bulk of fluid is initially contained in a closed trapezoidal reservoir. The reservoir is suspended over an elastic container, which is initially empty. Gravity is applied to the whole system. At time t_0, the bottom of the reservoir instantaneously opens, letting the fluid fall down under the action of gravity and eventually interact with the deformable container.

The geometrical parameters as well as the material properties of the fluid bulk and of the elastic container are reported in Table 20. Notice that for this problem the solid-to-fluid density ratio is extremely low ($\rho_s/\rho_f = 0.02$) and that the container is modeled as an elastic material with very low Young's modulus (21 MPa is in the range of Young's moduli of natural rubber), so that severe added-mass effects can be expected.

The problem is solved using both BGS with Aitken relaxation and IQN-ILS coupling strategies. The different FSI parameters employed are reported in Table 21.

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

Fig. 21. Cylinder pop-off — Contour of the fluid pressure field at six different instants in time for the IQN-ILS(30) case.



(a) $t = 0.1$s (b) $t = 0.15$s

(c) $t = 0.25$s (d) $t = 0.30$s

(e) $t = 0.35$s (f) $t = 1.0$s

Some snapshots of the evolution of the problem at different time steps, obtained using IQN-ILS(30) method, are reported in Fig. 24. As it can be observed, the fluid initially experiences a free fall (Fig. 24(a)), until it impacts the container at around t = 1.0 s. At the very beginning of the impact, the container stretches due to the force exerted by the fluid (Fig. 24(b)). A strongly coupled and complex interaction between the fluid and the container characterizes the problem from this moment on. The response

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

obtained by CUPyDO is qualitatively in very good agreement with the results of Meduri et al. [19] and Franci et al. [17]. An asymmetry develops over time, clearly visible at t = 3.5 s and t = 3.8 s, due to the intrinsic nonlinearity of the physical problem. Similar (yet different) lack of symmetry can be found elsewhere in the literature (see for instance Meduri et al. [19]).

Fig. 22. Cylinder pop-off — Time evolution No. of FSI iterations per time step. BGS vs IQN-ILS(30).

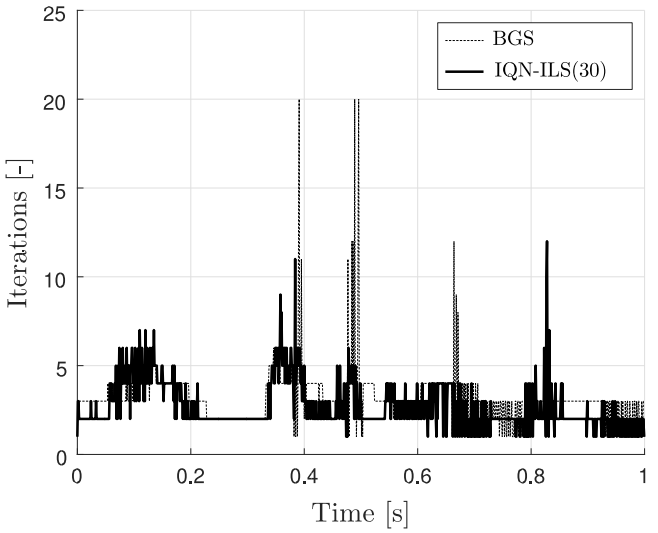
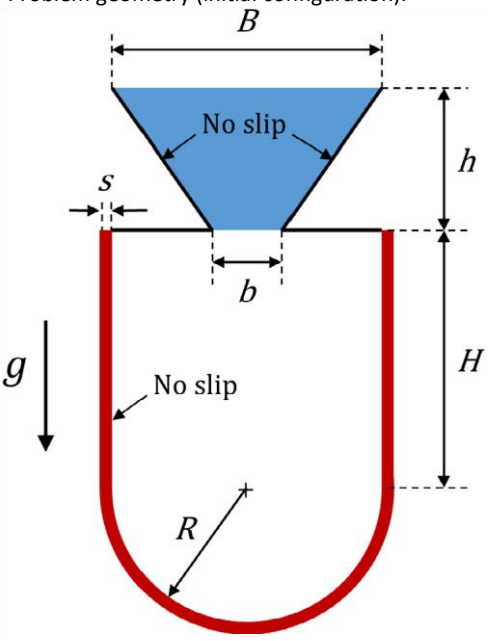


Fig. 23. Filling of an elastic container — Problem geometry (initial configuration).



Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

Table 20. Filling of an elastic container — Solid and fluid properties.

Geometry	Rigid container top side [m]	B	4.87
	Rigid container bottom side [m]	b	1.30
	Rigid container height [m]	h	2.50
	Elastic container straight side [m]	H	3.75
	Elastic container cap radius [m]	R	2.25
	Elastic container thickness [m]	s	0.20
	Gravity [m/s^2]	g	9.81
Solid	Density [kg/m^3]	ρ_s	20
	Young's modulus [Pa]	E	2.1×10^7
	Poisson's ratio [–]	ν_s	0.3
	No. of Linear Quad Elements		4×295
Fluid	Density [kg/m^3]	ρ_f	1000
	Dynamic viscosity [Pa · s]	μ_f	100
	No. of particles		5622

To obtain a more quantitative comparison, the time evolution of the vertical displacement of the bottom of the container is shown in Fig. 25, and compared to the results of Meduri et al. [19] and Franci et al. [17]. Globally, a very good agreement can be observed. In particular, the present solution and the one obtained by Meduri et al. practically coincide until $t = 3$ s, while the results obtained by Franci et al. present some differences and predict a lower value for the maximum displacement, registered around $t = 2.5$ s. Between $t = 3$ s and $t = 5$ s some differences appear between the three solutions. It is the authors' belief that this is mainly linked to the different formulations employed for the solid domain in the three works. In the end, from $t = 5$ s to $t = 10$ s, all three simulations predict very similar responses. In particular all seem to indicate the same value for the final displacement of the bottom of the container, which is around 0.35 m.

Table 21. Filling of an elastic container — FSI parameters.

Algorithm	BGS	IQN-ILS
Relaxation	Aitken	Static
ω_{max}	0.5	0.5
Filtering	–	Haelterman
Filtering tolerance	–	1×10^{-1}
No. of previous steps retained	–	5/10/30/50
Δt_{FSI} [s]	1×10^{-3}	1×10^{-3}
ε_{FSI} [m]	1×10^{-6}	1×10^{-6}

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

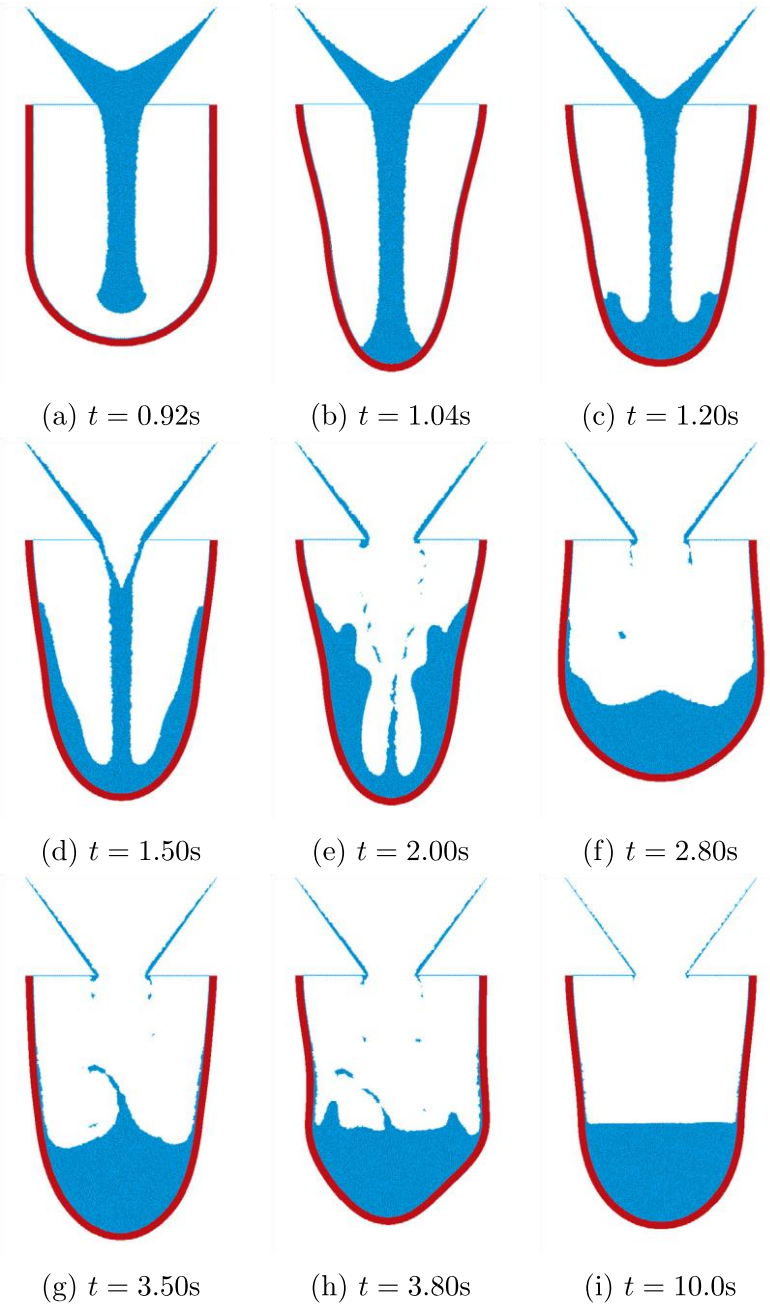
Table 22. Filling of an elastic container — Mean No. of FSI iterations.

Algorithm	Mean No. of iterations [–]
BGS	Does not converge
IQN-ILS(5)	7.94
IQN-ILS(10)	6.21
IQN-ILS(20)	5.33
IQN-ILS(30)	4.92
IQN-ILS(50)	4.93

The results presented above were obtained using the IQN-ILS(30) coupling. Indeed, the BGS coupling is not able to converge in this case, while the IQN-ILS(30) technique needs on average 4.92 iterations per time step to converge, as reported in Table 22. Table 22 also reports the average number of FSI iterations for the IQN-ILS(5), IQN-ILS(10) and IQN-ILS(50) cases. The conclusions drawn in example 5.4 in terms of mean number of FSI iterations are also confirmed in this case. Fig. 26 shows the time evolution of the number of iterations needed for convergence by BGS and IQN-ILS(30), respectively. As it can be observed, as soon as the fluid impacts the container, BGS starts experiencing convergence problems until the simulation is aborted (Fig. 26(b)). On the contrary, IQN-ILS(30) is able to simulate the entire 10 s of physical time without incurring convergence problems, as shown in Fig. 26(a).

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

Fig. 24. Filling of an elastic container — Snapshots of the simulation at different times. Results obtained using IQN-ILS(30).



Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

Fig. 25. Filling of an elastic container — Time evolution of the vertical displacement of the bottom of the container.

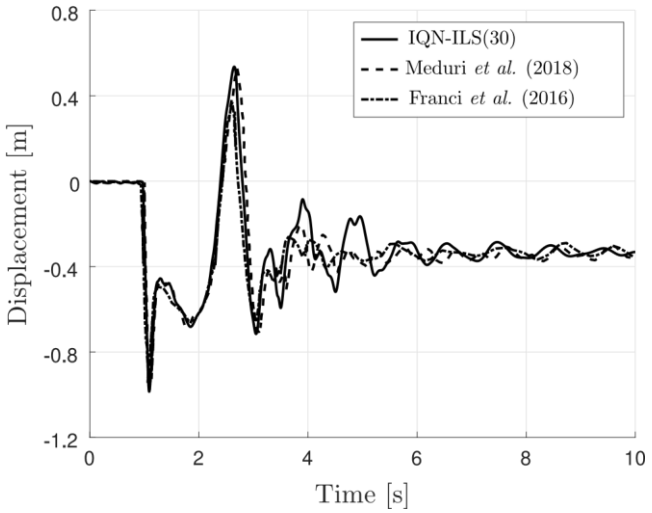
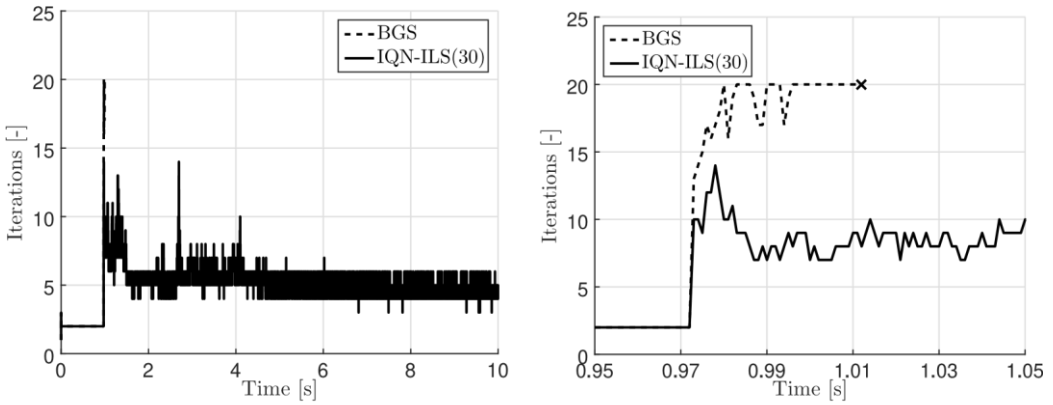


Fig. 26. Filling of an elastic container — Time evolution No. of FSI iterations. BGS vs IQN-ILS(30).



(a) Global (b) Zoom

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

6. Conclusions

In this work a fully partitioned Lagrangian framework for FSI problems characterized by the presence of free surfaces, very large solid deformations and displacements, and strong added-mass effects has been proposed. The solid was modeled through Metafor, a nonlinear Finite Element solver. The fluid was modeled using the Particle Finite Element Method (PFEM), a Lagrangian particle method that can handle extremely large deformations and designed to solve problems characterized by evolving interfaces, typically difficult to address with traditional CFD techniques. In particular, the potential of the approach, that relies on the PFEM to simulate the fluid and the Interface Quasi-Newton Inverse Least Squares (IQN-ILS) coupling strategy, for the solution of problems characterized by the presence of free surfaces and strong added-mass effects has been successfully demonstrated. A large number of benchmark tests for which added-mass effects are expected have been analyzed and the impact of different numerical parameters has been assessed. In terms of mean number of FSI iterations per time step, the IQN-ILS strategy has outperformed the classical Block Gauss–Seidel (BGS) method with Aitken relaxation for all tests performed with negligible additional cost. For instance, in the most extreme case, i.e. the filling of a very light and flexible elastic container with a viscous fluid (Section 5.6), BGS with Aitken relaxation was not able to provide a converged solution beyond the first contact between the fluid and the solid, while IQN-ILS converged without particular problems. Nonetheless, IQN-ILS requires the solution of an overdetermined problem and its convergence properties depend on the filtering employed. It has been shown that the filtering proposed by Haelterman et al. [70] is more efficient and robust than the filtering initially proposed by Degroote et al. [59,64], and should thus be favored.

Overall, it has been demonstrated that the PFEM, combined with the IQN-ILS strategy, can be successfully used in a fully partitioned fashion to solve a very large spectrum of FSI applications including large solid displacements and deformations, moving interfaces and free surfaces, and strong added-mass effects.

References

[1] Z. Li, J. Leduc, J. Nunez-Ramirez, A. Combescure, J.C. Marongiu, A non-intrusive partitioned approach to couple smoothed particle hydrodynamics and finite element methods for transient fluid–structure interaction problems with large interface motion, Comput. Mech. 55 (4) (2015) 697–718, http://dx.doi.org/10.1007/s00466-015-1131-8.

[2] C. Habchi, S. Russeil, D. Bougeard, J.L. Harion, T. Lemenand, A. Ghanem, D.D. Valle, H. Peerhossaini, Partitioned solver for strongly coupled fluid–structure interaction, Comput. & Fluids 71 (2013) 306–319, http://dx.doi.org/10.1016/j.compfluid.2012.11.004.

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

[3] W. Dettmer, D. Peric, A new staggered scheme for fluid–structure interaction, Internat. J. Numer. Methods Engrg. 93 (1) (2013) 1–22,´ http://dx.doi.org/10.1002/nme.4370.

[4] F. Nobile, C. Vergara, Partitioned algorithms for fluid–structure interaction problems in haemodynamics, Milan J. Math. 80 (2) (2012) 443–467, http://dx.doi.org/10.1007/s00032-012-0194-7.

[5] W.A. Wall, S. Genkinger, E. Ramm, A strong coupling partitioned approach for fluid–structure interaction with free surfaces, Comput. & Fluids 36 (1) (2007) 169–183, http://dx.doi.org/10.1016/j.compfluid.2005.08.007, Challenges and advances in flow simulation and modeling.

[6] U. Küttler, W.A. Wall, Fixed-point fluid–structure interaction solvers with dynamic relaxation, Comput. Mech. 43 (1) (2008) 61–72, http://dx.doi.org/10.1007/s00466-008-0255-5.

[7] C. Farhat, K.G. van der Zee, P. Geuzaine, Provably second-order time-accurate loosely-coupled solution algorithms for transient nonlinear computational aeroelasticity, Comput. Methods Appl. Mech. Engrg. 195 (17–18) (2006) 1973–2001, http://dx.doi.org/10.1016/j.cma.2004. 11.031.

[8] H.G. Matthies, J. Steindorf, Partitioned strong coupling algorithms for fluid–structure interaction, Comput. Struct. 81 (8–11) (2003) 805–812, http://dx.doi.org/10.1016/S0045-7949(02)00409-1.

[9] C.W. Hirt, B.D. Nichols, Volume of Fluid (VOF) method for the dynamics of free boundaries, J. Comput. Phys. 39 (1981) 201–225.

[10] S. Osher, R. Fedkiw, Level Set Methods and Dynamic Implicit Surfaces, Springer, 2006.

[11] S.R. Idelsohn, E. Oñate, F. Del Pin, The particle finite element method: a powerful tool to solve incompressible flows with free-surfaces and breaking waves, Internat. J. Numer. Methods Engrg. 61 (2004) 964–989.

[12] E. Oñate, S.R. Idelsohn, F. Del Pin, R. Aubry, The particle finite element method. An overview, Int. J. Comput. Methods 1 (2) (2004) 267–307.

[13] J.J. Monaghan, R.A. Gingold, Smoothed particle hydrodynamics: theory and application to non-spherical stars, Mon. Not. R. Astron. Soc. 181 (1977) 375–389.

[14] J.J. Monaghan, Smoothed particle hydrodynamics, Rep. Progr. Phys. 68 (2005) 1703–1759.

[15] T. Belytschko, Y.Y. Lu, L. Gu, Element-free Galerkin methods, Internat. J. Numer. Methods Engrg. 34 (2) (1994) 229–256.

[16] S.R. Idelsohn, J. Marti, A. Limache, E. Oñate, Unified Lagrangian formulation for elastic solids and incompressible fluids: Application to fluid–structure interaction problems via the PFEM, Comput. Methods Appl. Mech. Engrg. 197 (2008) 1762–1776.

[17] A. Franci, E. Oñate, J.M. Carbonell, Unified Lagrangian formulation for solid and fluid mechanics and FSI problems, Comput. Methods Appl. Mech. Engrg. 298 (2016) 520–547.

[18] M. Zhu, M.H. Scott, Improved fractional step method for simulating fluid–structure interaction using the PFEM, Internat. J. Numer. Methods Engrg. 99 (12) (2014) 925–944.

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

[19]S. Meduri, M. Cremonesi, U. Perego, O. Bettinotti, A. Kurkchubasche, V. Oancea, A partitioned fully explicit Lagrangian finite element method for highly nonlinear fluid–structure-interaction problems, Internat. J. Numer. Methods Engrg. 113 (2017) 43–64, http://dx.doi.org/ 10.1002/nme.5602.

[20]D. Thomas, M. Cerquaglia, R. Boman, T. Economon, J. Alonso, G. Dimitriadis, V. Terrapon, CUPyDO an integrated Python environment for coupled multi-physics simulations, Adv. Eng. Softw. 128 (2019) 69–85.

[21]H. Edelsbrunner, E.P. Mücke, Three-dimensional alpha shapes, ACM Trans. Graph. 13 (1) (1994) 43–72.

[22]A. Franci, M. Cremonesi, On the effect of standard PFEM remeshing on volume conservation in free-surface fluid flow problems, Comput. Part. Mech. 4 (3) (2016) 331–343.

[23]S.R. Idelsohn, E. Oñate, To mesh or not to mesh. That is the question..., Comput. Methods Appl. Mech. Engrg. 195 (2006) 4681–4696.

[24]M. Cremonesi, S. Meduri, U. Perego, A. Frangi, An explicit Lagrangian finite element method for free-surface weakly compressible flows, Comput. Part. Mech. 4 (3) (2016) 357–369, http://dx.doi.org/10.1007/s40571-016-0122-7.

[25]S.R. Idelsohn, M. Mier-Torrecilla, E. Oñate, Multi-fluid flows with the particle finite element method, Comput. Methods Appl. Mech. Engrg. 198 (2009) 2750–2767.

[26]S.R. Idelsohn, M. Mier-Torrecilla, N.M. Nigro, E. Oñate, On the analysis of heterogeneous fluids with jumps in the viscosity using a discontinuous pressure field, Comput. Mech. 46 (2010) 115–124.

[27]X. Zhang, K. Krabbenhoft, D.M. Pedroso, A.V. Lyamin, D. Sheng, M. Vicente da Silva, D. Wang, Particle finite element analysis of large deformation and granular flow problems, Comput. Geotech. 54 (2013) 133–142.

[28]M. Cremonesi, U. Perego, A Lagrangian finite element approach for the simulation of water-waves induced by landslides, Comput. Struct. 89 (2011) 1086–1093.

[29]S.R. Idelsohn, J. Marti, A. Souto-Iglesias, E. Oñate, Interaction between an elastic structure and free-surface flows: experimental versus numerical comparisons using the PFEM, Comput. Mech. 43 (1) (2008) 125–132, http://dx.doi.org/10.1007/s00466-008-0245-7.

[30]S.R. Idelsohn, F. Del Pin, R. Rossi, E. Oñate, Fluid–structure interaction problems with strong added-mass effect, Internat. J. Numer. Methods Engrg. 80 (2009) 1261–1294.

[31]E. Oñate, A. Franci, J.M. Carbonell, A particle finite element method for analysis of industrial forming processes, Comput. Mech. 54 (2014) 85–107.

[32]E. Oñate, R. Rossi, S.R. Idelsohn, K.M. Butler, Melting and spread of polymers in fire with the particle finite element method, Internat. J. Numer. Methods Engrg. 81 (2010) 1046–1072.

[33]S.R. Idelsohn, N.M. Nigro, J.M. Gimenez, R. Rossi, J.M. Marti, A fast and accurate method to solve the incompressible Navier–Stokes equations, Eng. Comput. 30 (2) (2013) 197–222, http://dx.doi.org/10.1108/02644401311304854.

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

[34]S.R. Idelsohn, J. Marti, P. Becker, E. Oñate, Analysis of multifluid flows with large time steps using the particle finite element method, Internat. J. Numer. Methods Fluids 75 (9) (2014) 621–644, http://dx.doi.org/10.1002/fld.3908.

[35]P.B. Ryzhakov, J. Marti, S.R. Idelsohn, E. Oñate, Fast fluid–structure interaction simulations using a displacement-based finite element model equipped with an explicit streamline integration prediction, Comput. Methods Appl. Mech. Engrg. 315 (2017) 1080–1097, http: //dx.doi.org/10.1016/j.cma.2016.12.003.

[36]M. Cerquaglia, G. Deliége, R. Boman, V. Terrapon, J.P. Ponthot, Free-slip boundary conditions for simulating free-surface incompressible flows through the particle finite element method, Internat. J. Numer. Methods Engrg. 110 (10) (2017) 921–946, http://dx.doi.org/10.1002/ nme.5439.

[37]Metafor, A nonlinear finite element code, University of Liège. http://metafor.ltas.ulg.ac.be/.

[38]T. Belytschko, W.K. Liu, B. Moran, Nonlinear Finite Elements for Continua and Structures, Wiley, 2001.

[39]I. Babuška, R. Narasimhan, The Babuška–Brezzi condition and the patch test: an example, Comput. Methods Appl. Mech. Engrg. 140 (1997) 183–199.

[40]F. Brezzi, M. Fortin, Mixed and Hybrid Finite Element Method, Springer, Berlin, 1991.

[41]T.E. Tezduyar, S. Mittal, S.E. Ray, R. Shih, Incompressible flow computations with stabilized bilinear and linear equal-order-interpolation velocity-pressure elements, Comput. Methods Appl. Mech. Engrg. 95 (1992) 221–242.

[42]M. Cremonesi, A. Frangi, U. Perego, A Lagrangian finite element approach for the analysis of fluid–structure interaction problems, Internat. J. Numer. Methods Engrg. 84 (5) (2010) 610–630.

[43]S.R. Idelsohn, E. Oñate, The challenge of mass conservation in the solution of free-surface flows with the fractional-step method: Problems and solutions, Int. J. Numer. Methods Biomed. Eng. 26 (2010) 1313–1330.

[44]P. Ryzhakov, E. Oñate, R. Rossi, S.R. Idelsohn, Improving mass conservation in simulation of incompressible flows, Internat. J. Numer. Methods Engrg. 90 (2012) 1435–1451.

[45]J.P. Ponthot, Unified stress update algorithms for the numerical simulation of large deformation elasto-plastic and elasto-viscoplastic processes, Int. J. Plast. 18 (1) (2002) 91–126, http://dx.doi.org/10.1016/S0749-6419(00)00097-8.

[46]J. Chung, G.M. Hulbert, A time integration algorithm for structural dynamics with improved numerical dissipation: The generalized-α method, J. Appl. Mech. 60 (2) (1993) 371–375.

[47]D.S. Malkus, T.J. Hughes, Mixed finite element methods — reduced and selective integration techniques: A unification of concepts, Comput. Methods Appl. Mech. Engrg. 15 (1) (1978) 63–81, http://dx.doi.org/10.1016/0045-7825(78)90005-1.

[48]J.C. Simo, M.S. Rifai, A class of mixed assumed strain methods and the method of incompatible modes, Internat. J. Numer. Methods Engrg. 29 (8) (1990) 1595–1638, http://dx.doi.org/10.1002/nme.1620290802.

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

[49]Q. Bui, L. Papeleux, J. Ponthot, Numerical simulation of springback using enhanced assumed strain elements, J. Mater Process. Technol. 153–154 (2004) 314–318, http://dx.doi.org/10.1016/j.jmatprotec.2004.04.342.

[50]L. Adam, J.P. Ponthot, Thermomechanical modeling of metals at finite strains: First and mixed order finite elements, Int. J. Solids Struct. 42 (21) (2005) 5615–5655, http://dx.doi.org/10.1016/j.ijsolstr.2005.03.020.

[51]J. Donea, A. Huerta, J.P. Ponthot, A. Rodríguez-Ferran, Arbitrary Lagrangian–Eulerian methods, in: Encyclopedia of Computational Mechanics, John Wiley & Sons, 2004, Ch. 14.

[52]R. Boman, J.P. Ponthot, Efficient ALE mesh management for 3D quasi-Eulerian problems, Internat. J. Numer. Methods Engrg. 92 (10) (2012) 857–890, http://dx.doi.org/10.1002/nme.4361.

[53]R. Koeune, J.P. Ponthot, A one phase thermomechanical model for the numerical simulation of semi-solid material behavior. Application to thixoforming, Int. J. Plast. 58 (2014) 120–153, http://dx.doi.org/10.1016/j.ijplas.2014.01.004.

[54]P.P. Jeunechamps, J.P. Ponthot, An efficient 3D implicit approach for the thermomechanical simulation of elastic-viscoplastic materials submitted to high strain rate and damage, Internat. J. Numer. Methods Engrg. 94 (10) (2013) 920–960, http://dx.doi.org/10.1002/nme.4489.

[55]M. Mengoni, J.P. Ponthot, Isotropic continuum damage/repair model for alveolar bone remodeling, J. Comput. Appl. Math. 234 (7) (2010) 2036–2045, http://dx.doi.org/10.1016/j.cam.2009.08.061.

[56]R. Boman, J.P. Ponthot, Finite element simulation of lubricated contact in rolling using the arbitrary Lagrangian–Eulerian formulation, Comput. Methods Appl. Mech. Engrg. 193 (39) (2004) 4323–4353, http://dx.doi.org/10.1016/j.cma.2004.01.034.

[57]C. Farhat, M. Lesoinne, P.L. Tallec, Load and motion transfer algorithms for fluid/structure interaction problems with non-matching discrete interfaces: Momentum and energy conservation, optimal discretization and application to aeroelasticity, Comput. Methods Appl. Mech. Engrg. 157 (1) (1998) 95–114, http://dx.doi.org/10.1016/S0045-7825(97)00216-8.

[58]P. Causin, J. Gerbeau, F. Nobile, Added-mass effect in the design of partitioned algorithms for fluid–structure problems, Comput. Methods Appl. Mech. Engrg. 194 (42–44) (2005) 4506–4527, http://dx.doi.org/10.1016/j.cma.2004.12.005.

[59]J. Degroote, K.J. Bathe, J. Vierendeels, Performance of a new partitioned procedure versus a monolithic procedure in fluid–structure interaction, Comput. Struct. 87 (11–12) (2009) 793–801, http://dx.doi.org/10.1016/j.compstruc.2008.11.013.

[60]J. Vierendeels, L. Lanoye, J. Degroote, P. Verdonck, Implicit coupling of partitioned fluid–structure interaction problems with reduced order models, Comput. Struct. 85 (11–14) (2007) 970–976, http://dx.doi.org/10.1016/j.compstruc.2006.11.006.

[61]H.J. Bungartz, F. Lindner, B. Gatzhammer, M. Mehl, K. Scheufele, A. Shukaev, B. Uekermann, preCICE — a fully parallel library for multi-physics surface coupling, Comput. & Fluids 141 (2016) 250–258, http://dx.doi.org/10.1016/j.compfluid.2016.04.003, Advances in fluid–structure interaction.

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

[62]A. Beckert, H. Wendland, Multivariate interpolation for fluid–structure-interaction problems using radial basis functions, Aerosp. Sci. Technol. 5 (2) (2001) 125–134, http://dx.doi.org/10.1016/S1270-9638(00)01087-7.

[63]A. de Boer, A. van Zuijlen, H. Bijl, Review of coupling methods for non-matching meshes, Comput. Methods Appl. Mech. Engrg. 196 (8) (2007) 1515–1525, http://dx.doi.org/10.1016/j.cma.2006.03.017, Domain decomposition methods: recent advances and new challenges in engineering.

[64]J. Degroote, A. Souto-Iglesias, W.V. Paepegem, S. Annerel, P. Bruggeman, J. Vierendeels, Partitioned simulation of the interaction between an elastic structure and free surface flow, Comput. Methods Appl. Mech. Engrg. 199 (33–36) (2010) 2085–2098, http://dx.doi.org/10.1016/ j.cma.2010.02.019.

[65]M.M. Joosten, W.G. Dettmer, D. Peric, Analysis of the block Gauss–Seidel solution procedure for a strongly coupled model problem with´ reference to fluid–structure interaction, Internat. J. Numer. Methods Engrg. 78 (7) (2009) 757–778, http://dx.doi.org/10.1002/nme.2503.

[66]C. Wood, A. Gil, O. Hassan, J. Bonet, Partitioned block-Gauss–Seidel coupling for dynamic fluid–structure interaction, Comput. Struct. 88 (23–24) (2010) 1367–1382, http://dx.doi.org/10.1016/j.compstruc.2008.08.005, Special issue: Association of Computational Mechanics — United Kingdom.

[67]R. Rossi, E. Oate, Analysis of some partitioned algorithms for fluid–structure interaction, Eng. Comput. 27 (1) (2010) 20–56, http: //dx.doi.org/10.1108/02644401011008513.

[68]B.M. Irons, R.C. Tuck, A version of the Aitken accelerator for computer iteration, Internat. J. Numer. Methods Engrg. 1 (3) (1969) 275–277, http://dx.doi.org/10.1002/nme.1620010306.

[69]J. Degroote, P. Bruggeman, R. Haelterman, J. Vierendeels, Stability of a coupling technique for partitioned solvers in FSI applications, Comput. Struct. 86 (23–24) (2008) 2224–2234, http://dx.doi.org/10.1016/j.compstruc.2008.05.005.

[70]R. Haelterman, A. Bogaers, K. Scheufele, B. Uekermann, M. Mehl, Improving the performance of the partitioned QN-ILS procedure for fluid–structure interaction problems: Filtering, Comput. Struct. 171 (Supplement C) (2016) 9–17, http://dx.doi.org/10.1016/j.compstruc. 2016.04.001.

[71]C. Förster, W.A. Wall, E. Ramm, Artificial added mass instabilities in sequential staggered coupling of nonlinear structures and incompressible viscous flows, Comput. Methods Appl. Mech. Engrg. 196 (7) (2007) 1278–1293, http://dx.doi.org/10.1016/j.cma.2006.09.002.

[72]J. Degroote, S. Annerel, J. Vierendeels, Stability analysis of Gauss–Seidel iterations in a partitioned simulation of fluid–structure interaction, Comput. Struct. 88 (5–6) (2010) 263–271, http://dx.doi.org/10.1016/j.compstruc.2009.09.003.

[73]M. Olivier, J.F. Morissette, G. Dumas, A fluid–structure interaction solver for nano-air-vehicle flapping wings, in: Fluid Dynamics and Co-located Conferences, American Institute of Aeronautics and Astronautics, 2009, http://dx.doi.org/10.2514/6.2009-3676.

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

[74]W. Dettmer, D. Peric, A computational framework for fluid–structure interaction: Finite element formulation and applications, Comput.´ Methods Appl. Mech. Engrg. 195 (41–43) (2006) 5754–5779, http://dx.doi.org/10.1016/j.cma.2005.10.019, John H. Argyris memorial issue. Part II.

[75]F. Palacios, M. Colonno, A. Aranake, A. Campos, S. Copeland, T. Economon, A. Lonkar, T. Lukaczyk, T. Taylor, J. Alonso, Stanford University Unstructured (SU2): An open-source integrated computational environment for multi-physics simulation and design, in: AIAA 51st Aerospace Sciences Meeting, Grapevine, TX, 7–10 January, 2013. http://dx.doi.org/10.2514/6.2013-287.

[76]T.C. Papanastasiou, N. Malamataris, K. Ellwood, A new outflow boundary condition, Internat. J. Numer. Methods Fluids 14 (5) (1992) 587–608.

[77]D.F. Griffiths, The 'no boundary condition' outflow boundary condition, Internat. J. Numer. Methods Fluids 24 (4) (1997) 393–411.

[78]M. Renardy, Imposing 'no' boundary condition at outflow: Why does it work? Internat. J. Numer. Methods Fluids 24 (4) (1997) 413–417.

[79]E. Walhorn, A. Kölke, B. Hübner, D. Dinkler, Fluid–structure coupling within a monolithic model involving free surface flows, Comput. Struct. 83 (2005) 2100–2111.

[80]M. Liu, J. Shao, H. Li, Numerical simulation of hydro-elastic problems with smoothed particle hydrodynamics method, J. Hydrodynamics 25 (5) (2013) 673–682.

[81]P.B. Ryzhakov, R. Rossi, S.R. Idelsohn, E. Oate, A monolithic Lagrangian approach for fluid–structure interaction problems, Comput. Mech. 46 (2010) 883–899.

[82]A. Rafiee, K.P. Thiagarajan, An SPH projection method for simulating fluid-hypoelastic structure interaction, Comput. Methods Appl. Mech. Engrg. 198 (33–36) (2009) 2785–2795, http://dx.doi.org/10.1016/j.cma.2009.04.001.

[83]S. Meduri, M. Cremonesi, U. Perego, A fully explicit fluid-structure interaction approach based on the PFEM, in: VII International Conference on Computational Methods for Coupled Problems in Science and Engineering, 2017, pp. 299–306.

[84]C. Antoci, M. Gallati, S. Sibilla, Numerical simulation of fluid–structure interaction by SPH, Comput. Struct. 85 (11–14) (2007) 879–890, http://dx.doi.org/10.1016/j.compstruc.2007.01.002.

[85]Q. Yang, V. Jones, L. McCue, Free-surface flow interactions with deformable structures using an SPH-FEM model, Ocean Eng. 55 (Supplement C) (2012) 136–147, http://dx.doi.org/10.1016/j.oceaneng.2012.06.031.

[86]C. Antoci, Simulazione numerica dell'interazione fluido-struttura con la tecnica SPH (Ph.D. thesis), Università degli studi di Pavia, 2006.

[87]C. Hesch, A. Gil, A.A. Carreo, J. Bonet, On continuum immersed strategies for fluid–structure interaction, Comput. Methods Appl. Mech. Engrg. 247–248 (2012) 51–64, http://dx.doi.org/10.1016/j.cma.2012.07.021.

Published in: Computer Methods in Applied Mechanics and Engineering (2019), vol. 348, pp. 409–442
DOI: 10.1016/j.cma.2019.01.021
Status: Postprint (Author's version)

[88]A. Gil, A.A. Carreo, J. Bonet, O. Hassan, The immersed structural potential method for haemodynamic applications, J. Comput. Phys. 229 (22) (2010) 8613–8641, http://dx.doi.org/10.1016/j.jcp.2010.08.005.

[89]X. Wang, W.K. Liu, Extended immersed boundary method using FEM and RKPM, Comput. Methods Appl. Mech. Engrg. 193 (12) (2004) 1305–1321, http://dx.doi.org/10.1016/j.cma.2003.12.024, Meshfree methods: Recent advances and new applications.