An embedded Finite Element framework for the resolution of strongly coupled Fluid-Structure Interaction problems. Application to volumetric and membrane-like structures.

R. Zorrilla[a,b,*], R. Rossi[1,1], R. Wüchner[1,1], E. Oñate[1,1]

[a] *Universitat Politècnica de Catalunya (UPC), Departament d'Enginyeria Civil i Ambiental*
[b] *International Center for Numerical Methods in Engineering (CIMNE)*
[c] *Technische Universität München, Lehrstuhl für Statik*

Abstract

This work presents a Fluid-Structure Interaction framework for the robust and efficient simulation of strongly coupled problems involving arbitrary large displacements and rotations. We focus on the application of the proposed tool to lightweight membrane-like structures. Nonetheless, all the techniques we present in this work can be applied to both volumetric and volumeless bodies. To achieve this, we rely on the use of embedded mesh methods in the fluid solver to conveniently handle the extremely large deflections and eventual topology changes of the structure. The coupling between the embedded fluid and mechanical solvers is based on an interface residual black-box strategy. We validate our proposal by solving reference benchmarking examples that consider both volumetric and volumeless geometries. Whenever it is possible, we also compare the embedded solution with the one obtained with our reference body fitted solver. Finally we present a real-life application of the presented embedded Fluid-Structure Interaction solver.

Keywords: Fluid-Structure Interaction, Embedded Boundary Methods, Level set methods, Coupled problems, Black-box coupling, Volumeless bodies

1. Introduction

1.1. The Fluid-Structure Interaction problem. Numerical approaches.

The understanding of the interaction mechanisms that occur between a moving body and a surrounding (or passing through) fluid was one of the most relevant engineering challenges of the past century. Initially, the interest in this

*Corresponding author
Email addresses: rzorrilla@cimne.upc.edu (R. Zorrilla), rrossi@cimne.upc.edu (R. Rossi), wuechner@tum.de (R. Wüchner), onate@cimne.upc.edu (E. Oñate)

phenomenon, which is widely known as Fluid-Structure Interaction (FSI), was limited to the comprehension and quantification of the dynamic effects arising from the interaction with unsteady fluid flows. Some practical examples are the aeroelastic phenomena (e.g. fluttering) occurring in slender structures such as aerofoils or bridge decks [1], the performance analysis of wind mill blades [2] or the water hammer in fluid ducts.

There also exist lots of natural phenomena, which include some of the human body processes, that involve FSI mechanisms. Specially in recent decades, this has motivated the appearance of new research lines that aim at understanding the mechanics behind them. Some examples are the flapping of insects wings [3], the hydrodynamics of fish fins [3] or the motion of jellyfish [4]. In the biomedical field, we can find some successful applications in [5], [6], [7] and [8], which study the blood flow in human vessels. Similarly, FSI numerical methods are applied in [9], [10] and [11] to study the movement of the hearth valves or, more recently, in [12] to study the human cell motion.

The main feature common to all FSI problems is the mutual dependency between the Computational Fluid Dynamics (CFD) and the Computational Solid Mechanics (CSM) problems. Such dependency comes from the fact that the structure is deformable under the action of the fluid load. At the same time, the structure deformation implies a modification in the fluid problem geometry. This turns into a change in the load distribution over the structure, meaning that the fluid and the mechanical problem are somehow linked. This makes the FSI to belong to the so called coupled family of problems.

In this context, the tracking of the wet (or coupling) interface between the fluid and structure domains becomes crucial for the proper resolution of the FSI problem. Based on the movement of such coupling interface, we can roughly divide the FSI problems in two main groups.

On the one hand, it may happen that there is barely interaction between the fluid and the solid domains (e.g. massive civil engineering structures). In these cases, which are denoted as *one-way coupled* FSI problems, it is commonly assumed that there is no interaction on the wet interface. This greatly simplifies the problem as the coupling becomes in a simple load mapping from the fluid to the structure.

At the other extreme, we find the *strongly coupled* FSI problems, whose solution involves a high dependency between the fluid and the structure subdomain problems. This implies the need of taking into account arbitrary, and possibly large, movement of the wet interface.

This is not a problem in the solid domain as it is generally solved in a Lagrangian frame of reference. On the contrary, standard Eulerian fluid techniques are no longer valid as they cannot consider the movement of the coupling interface. Traditionally, this has been achieved either by upgrading the Eulerian frame of reference to the Arbitrary Lagrangian Eulerian (ALE) one [13][14] or by using non-conforming approaches such as the Immersed Boundary Method (IBM) [15].

The ALE methods are based on modifying the Eulerian advection velocity such that the solution is still consistent despite the movement of the Eulerian

mesh. As the movement of the background mesh is only known in the coupling interface and domain boundaries, it is required to solve an extra mesh motion problem to spread it into the domain. The most common approach to achieve this is the Laplacian smoothing [16]. Alternatively, one can solve a structural similarity problem [17], which is known to be slightly more robust, specially when moderate rotations appear.

Disregarding the extra mesh motion problem overhead, the main disadvantage of ALE methods is that they are prone to yield excessively distorted, or even inverted, elements when large displacements or rotations occur in the boundaries. As a consequence, remeshing is often needed to tackle these undesired situations.

Another option to deal with moving interfaces, is to adopt purely Lagrangian approaches such as the Particle Finite Element Method (PFEM) [18][19][20]. While the use of a Lagrangian framework provides obvious advantages in following the internal interfaces, such methods imply remeshing at every time step.

Chimera type methods are also a feasible alternative for the problems at hand. These are based on representing the analysed bodies with a set of independent body conforming mesh patches, that can freely move over the background fluid domain. Thanks to this feature the rigid body movements can be easily treated. However, the method might still suffer from local elemental degeneration due to the deformation of the patch meshes.

All the methods that we have presented so far are based on body conforming (also known as body fitted) discretizations. This means that the mesh boundaries match, as closely as possible, the ones of the analysed bodies. Even though this is the most straightforward representation of the geometry, which typically leads to a simplification in the Boundary Conditions (BCs) imposition, it comes at the price of requiring a mesh updating technique (mesh motion or remeshing).

1.2. Embedded mesh methods in Fluid-Structure Interaction problems

1.2.1. Embedded computational fluid dynamics

The need to overcome some of the limitations of body fitted approaches has led to the development of a wide variety of non-conforming mesh methods. Some of these are the IBM [15][21], the Embedded Boundary Method (EBM) (also known as Cut-FEM) [22] and the Shifted Boundary Method (SBM) [23][24].

All these techniques feature a volume mesh discretization, denoted as volume or background mesh, that is completely independent of the analysed bodies. As the interaction boundaries are no longer represented by the mesh, these techniques require doing some extra operations to implicitly represent the FSI boundaries in the background mesh.

Although this introduces another difficulty to the problem, the implicit representation becomes in an advantage when mesh updating issues are experienced since it allows substituting the mesh deformation problem by an update of the objects representation.

We highlight [25] as the pioneering work that applied non-conforming mesh methods to the resolution of an FSI problem. To consider the effect that the

immersed body has in the fluid, the authors apply an artificial body force. This same approach is successfully applied also in [15] and [26]. However, it is reported to lack accuracy and (or) stability in other applications.

As an alternative, approaches based on the level set method can be used [27]. These implicitly represent the analysed geometries by using a signed distance function, which is computed by an auxiliary algorithm. Although all these algorithms are built on top of the skin and background mesh intersections calculation, we can classify them into two main families.

On the one hand, there are the algorithms that deal with the concept of inside/outside by using computer graphics techniques such as the ray-casting. These are the most extended level set methods as they always return a smooth continuous level set function (Fig. 1). Their main drawback is that they can only be applied to volumetric bodies. This is a direct consequence of the fact that the distance is computed node-by-node.

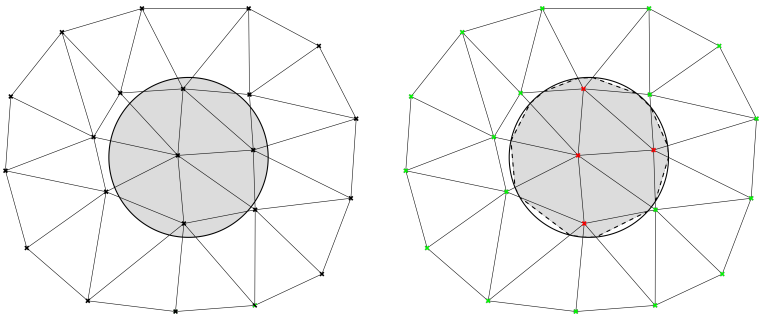


Figure 1: Continuous distance function. Body with a well defined internal volume (left) and its continuous distance representation (right). Green and red markers denote the positive and negative distance nodes. The dashed lines represent the zero isosurface ´skin representation (source [28]).

On the other hand, we have the discontinuous level set algorithms [29]. As it can be observed in Fig. 2, in this case the level set is not computed node-by-node but element-by-element. This is the key aspect that makes possible to represent not only volumetric bodies, but also membrane-like bodies with no internal volume [30][28]. Nonetheless, this feature comes at the price of having a potentially discontinuous level set, meaning that the same node may have different distance signs and values depending on the element considered.

Among all the level set based non-conforming mesh methods, the IBM is possibly the most straightforward approach to deal with the problems at hand. In short, the IBM consists in using the first layer of nodes laying inside the analysed bodies to impose the fluid essential BCs [31]. This means that the BCs are not imposed on the wet interface but on its closest dry nodes, something that makes the method prone to suffering from mass conservation issues. Nevertheless, it is possible to find different flavours of the IBM that are successfully applied to solve challenging FSI problems [32][33][34].

In this context, the SBM [23][24] arises as a novel alternative that aims to

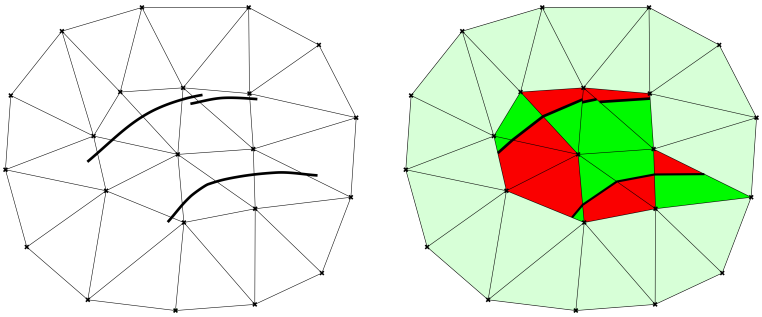


Figure 2: Discontinuous distance function. Body without internal volume (left) and its discontinuous distance representation (right). Red and green portions of the cut elements indicate the positive and negative discontinuous distance regions. Light green denotes the non-intersected elements (source [28]).

overcome the potential mass conservation issues of the IBM. This is achieved by computing a modified boundary value that takes into account the distance between the level set and the dry nodes where the BCs are imposed.

Both the IBM and the SBM rely on the internal (dry) nodes to impose the fluid BCs. This precludes the analysis of thin-walled (volumeless) structures as the inside/outside concept becomes meaningless in this case.

In this regard, the eXtended Finite Element Method (X-FEM) appears a feasible alternative to the previous methods. As the boundary condition is imposed over the level set zero isosurface, the X-FEM can be applied to both CFD [35] and FSI [36][37] problems involving not only volumetric structures but also thin-walled ones. The most representative feature of the X-FEM is that the FE space is enriched in the elements intersected by the embedded skin. Such enhanced FE space makes possible the representation of the velocity and pressure discontinuities arising after the immersion of the analysed bodies. However, this comes at the cost of introducing extra Degrees Of Freedom (DOFs) that correspond to the enrichment unknowns.

The extra DOFs computational overhead becomes crucial when the problem involves moving boundaries as the system matrix graph needs to be reconstructed each time the level set is updated. Besides this, the extension of the method to distributed memory environments could be a challenge due to the presence of blending elements, which might lay in a different partition that the one of the intersected element.

The EBM is another feasible alternative to deal with the problems at hand. Like the X-FEM, the EBM also relies on applying the BCs over the Solid-Fluid interface cuts. Such BCs imposition is, however, done in a weak sense by using techniques such as the penalty method, the Nitsche method [38] or Lagrange multipliers based methods [39]. Despite the weak BCs imposition may introduce some stability issues, the EBM is purely local, avoiding thus the previously commented computational drawbacks of the X-FEM.

Among all the weak BCs imposition techniques we prefer the Nitsche-based

approaches, as they are known to be more stable than penalty based ones. Moreover, they avoid the *inf-sup* stability issues and the DOFs set modification (and matrix graph reconstruction) of Lagrange multiplier methods.

It is possible to find in the literature multiple Nitsche based approaches to enforce the no-slip (stick) condition over the embedded fluid interface [40][41][42]. We highlight the approach presented in [40] due to its simplicity and performance. Concerning the slip condition, the Nitsche method is also used in [43] and [44], which we also highlight because of its stability and wide application range. However, all these techniques can only be applied to bodies that have a well defined internal volume.

In [3] this limitation is overcome by using a ghost cell method that relies on the neighbouring elements to enforce the embedded BCs. In [28] a purely elemental alternative is presented. This approach is based on the substitution of the common FE space within the intersected elements by a discontinuous one proposed in [45].[1] This makes possible to represent the velocity and pressure discontinuities arising from the immersion of any body, regardless of its type (volumetric or volumeless). This formulation is enhanced in [46] by using a Nitsche method to impose a Navier-slip condition, allowing thus the representation of any wall behaviour from the no-slip to the slip limits.

1.2.2. Embedded mesh methods and moving bodies

As commented before, one motivation for the use of non-conforming mesh methods in the context of FSI is that they can efficiently handle arbitrary large movements and rotations. This enhanced robustness, however, comes at the price of some particularities that require to be taken care of.

The first one is the small cut instability which manifests when the level set function cuts an element so that there is a very small intersection. Although this issue also happens when solving steady objects CFD problems, its frequency may be even higher when the analysed bodies move across the background mesh. Therefore, it is required to use an auxiliary strategy to ensure the convergence of the problem. Some approaches, such as the ghost penalty method [22] or the ghost cell method [47], prevent the small cut ill-conditioning by using the neighbouring nodes or elements to control the solution.

Although the stability of this family of methods is proven, to the best of our knowledge no purely local approach exists that does not require the neighbouring elements calculation. Taking this into account, our proposal to avoid the small cut instabilities is to use a level set quality check and correction algorithm [28]. The robustness of this approach is evinced in [28], where the distance check and correction algorithm is applied in the resolution of CFD problems involving potentially ill-conditioned distance functions.

The second particularity is the historical data initialization problem. This issue appears when the values required for the temporal derivatives approxima-

[1] From now on we will denote this modified discontinuous FE space as Ausas FE space after the name of the original author.

tion are not consistently initialized in those nodes that move from one side to the other of the level set (Fig. 3). As reported in [28], this leads to an unphysical velocity field in the surroundings of the moving body as the approximation of the inertial terms is not consistent with the updated level set position. We remark that no distinction is done between continuous and discontinuous distance functions, as the historical values need to be consistently updated in both cases.

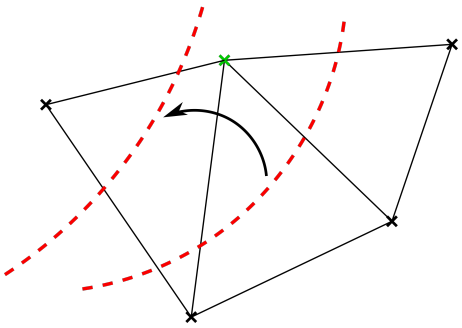


Figure 3: Embedded nodal initialization example. The node highlighted in green changes its position from one side to the other of the level set (red dashed line), which moves as indicated by the black arrow. This requires to re-initialize the historical values in the green node to consistently approximate the time derivatives.

For the consistent historical data initialization we rely on the Fixed Mesh - Arbitrary Lagrangian Eulerian (FM-ALE) method. The FM-ALE method is described and applied to moving boundaries embedded CFD problems in [48]. This is extended in [49], where it is similarly applied to solid mechanics and FSI problems.

The main idea behind the FM-ALE method is to retrieve the consistent historical values from an auxiliary mesh, which is denoted as virtual mesh by the original authors. Such virtual mesh, is initially set as a copy of the background one. Then it is deformed by solving a mesh moving problem to follow the motion of the level set. After doing this, the historical values, as well as the mesh velocity, are projected back from the virtual mesh to the background origin one where the embedded CFD problem is solved. Finally, the virtual mesh is reset to its original configuration.

We note that the FM-ALE mesh motion problem likely involves small displacements and rotations as the virtual mesh movement is not accumulated but restarted at each time the FM-ALE algorithm is applied. As a consequence, the common problems associated with body conforming ALE solvers (element distortion or inversion) rarely appear when using the FM-ALE method. This is also confirmed in [48], [49] and [50] where the FM-ALE method is applied to problems that involve large displacements and rotations of the immersed bodies.

1.3. Algorithms for Fluid-Structure Interaction

1.3.1. Coupling schemes: FSI transmission conditions

The resolution of any coupled problem requires satisfying the so called interface transmission conditions. In the FSI case, this implies complying with the interface force equilibrium and continuity. In order to achieve this, a coupling scheme is required.

Coupling schemes can be classified according to how the transmission conditions are enforced in: Dirichlet-Neumann (DN), Neumann-Neumann (NN) and the Robin type ones such as the Robin-Robin (RR) or the Neumann-Robin (NR). In this work we opt for a DN coupling scheme due to its proved performance and stability. Besides, its implementation is reasonably simple and does not require to modify the subdomain formulations, enabling thus the use of non-intrusive coupling algorithms.

Hence, a Dirichlet BC is enforced over the coupling interface of one subdomain. This is commonly the lower density one, which in the FSI case is most likely the fluid medium. Accordingly, the fluid load is applied as a Neumann BC over the structure interface. The stability and performance of the DN scheme, as well as its interaction with different time discretization schemes, are thoroughly studied in [51].

Nevertheless, we note that DN schemes may lack convergence or have stability issues in some particular applications. In this context, NN and Robin type schemes can be an effective choice to overcome these limitations. As these applications are out of the scope of this work, we refer the reader to [52] and [53] (RR) as well as to [54] and [55] (NR) for a detailed analysis of these alternative coupling schemes.

1.3.2. Coupling strategy: monolithic and partitioned coupling

After defining the transmission conditions that ensure the coupling between the subdomain solutions, it is required to define a coupling strategy to solve the problem. These can be divided in two main families: monolithic and partitioned (also known as staggered) strategies.

Monolithic strategies are considered to be the more robust option to solve a coupled problem. They consist in solving all the subdomain problems within a unique coupled system of equations. This approach is successfully applied to FSI problems in [19] and [56]. More related to our case, in [49] the authors use an embedded monolithic formulation to solve both the fluid and the structure domains.

Despite its better coupling stability, monolithic strategies tend to yield poorly conditioned system matrices that eventually preclude the use of iterative solvers. Moreover their implementation is intrusive and typically requires the development of a new specific solver for the coupling.

On the contrary, partitioned approaches keep a separated solver for each one of the subdomains. This enables the reuse of already existent robust and widely validated subdomain solvers, thus allowing to focus the implementation effort on the coupling algorithm. However, this advantage comes at the price of a convergence rate that is typically not as good as that of monolithic alternatives.

Another feature of partitioned strategies is that the fulfilment of the transmission conditions is ensured by exchanging information at the coupling interfaces. Depending on how many times per time step such information exchange occurs, we can further classify these in loosely-coupled (or explicit) and strongly-coupled partitioned strategies.

In loosely-coupled staggered approaches the information exchange occurs once per time step [57]. Although these is the cheapest approach in terms of computational and implementation effort, they only converge when there is limited interaction between the coupled subdomains. Besides, they are known to suffer a lot from the so called added mass effect instability that appears when solving FSI problems involving incompressible fluids [58][59].

Conversely, in strongly-coupled staggered approaches the information exchange occurs until a certain coupling convergence criterion is reached. Even though the convergence rate is not optimal, implicit staggered approaches can be used in combination with convergence acceleration algorithms to alleviate this when dealing with strongly coupled problems. Concerning the computational effort, as pointed in [60] it is difficult to do a fair comparison with monolithic approaches because this is completely problem dependent.

Besides, according to how the information is exchanged we can further classify implicit strategies in Jacobi and Gauss-Seidel type iterations. In Jacobi type strategies all the information exchanges occur at the same time, allowing thus the parallel resolution of the subdomain problems. Contrariwise, Gauss-Seidel type strategies solve the subdomain problems in a sequential manner. This means that the latest obtained data from one subdomain is used within the same iteration to solve the next one, as opposite to Jacobi type iterations, which will use the latest information in the next iteration.

Furthermore, there also exist alternative semi-implicit strategies. To this regard, we refer the reader to [55] where a comparative study of multiple alternatives to traditional coupling strategies can be found.

After having reviewed all the possible coupling strategies, we opt to use a Gauss-Seidel strongly-coupled staggered strategy. Disregarding monolithic approaches, this is the most robust technique that allows to reuse our existent CSM and embedded CFD solvers. At the same time, it brings up the concepts of black-box coupling [61][62]. Hence, we also require our coupling algorithm to not interfere with the subdomain solvers more than to get and set information, implying that these are considered as black boxes that return a solution for the provided input data.

Finally, we would like to highlight that the use of black-box coupling techniques not only enables the use of already existent in house codes but also external, even commercial, ones. This new wave of flexible coupling is denoted by the community as co-simulation [63].

1.3.3. Black-box interface residual minimization techniques

The selection of a Gauss-Seidel strongly-coupled strategy requires the definition of an interface residual function to be minimized. By doing this, it is ensured that the coupling transmission conditions are fulfilled. Since the choice

of such residual is completely problem dependent, in this subsection we assume that it is already defined. We recover this in the methodology chapter 3 in where we discuss the particularities of the embedded FSI interface residual calculation.

As we are in favour of black-box coupling techniques, we require our residual minimization technique to have restricted access to the subdomain solvers.

The most straightforward approach is a fixed-point iteration with relaxation. Despite convergence can be reached using a constant relaxation parameter, the computational cost is normally prohibitive. In this context, dynamic relaxation schemes appear as a rather simple but effective alternative to reduce the number of coupling iterations. Among all of them, we highlight the second order Aitken relaxation scheme, which is widely used in the FSI community [64][65]. Although the Aiken relaxation scheme has an astonishing performance compared to its implementation effort, the required iterations are still far from being optimal.

A more efficient alternative is to pose the interface residual problem as a non-linear Newton-Raphson (N-R) iterative procedure, leading to the so called Jacobian-based resolution schemes. However, the use of a black-box coupling scheme intentionally precludes the exact calculation of the interface Jacobian that minimizes the problem. Nonetheless, this can be avoided by using Jacobian-free Newton-Krylov (JFNK)[66] or Quasi-Newton (QN) methods [62].

An application of JFNK methods in the FSI context can be found in [67]. Rather than approximating the complete interface Jacobian, JFNK methods are based on approximating its projection onto the iteration update vector by using a finite differences formula. Due to the inherent non-linearity of the FSI problem, such projection needs to be linearized by introducing a small perturbation in the finite differences formula. Although there exist in the literature some formulas to compute the value of such small perturbation [68], its optimal value is completely problem dependent.

The selection of such perturbation constant can be bypassed by using a QN algorithm. In short, the idea behind any QN algorithm is to use the information from previous iterations to calculate an approximation of the interface Jacobian by using a linearized formula. Depending on how such formula is obtained, we end up with one or another QN method. In the following we highlight those ones that have been successfully applied to solve FSI problems.

Although nowadays it is not the most popular method, we acknowledge Broyden's iteration as the pioneer QN algorithm applied to FSI problems [68]. A more recent approach is the Interface Quasi-Newton with Inverse Jacobian from Least Squares model (IQN-ILS) proposed by Degroote et. al. in [60]. The IQN-ILS is further studied in [69], where the authors do a thorough performance analysis of the method, showing that the IQN-ILS performance overcomes the Aitken and JFNK ones in all the reported examples. A fairly similar approach to the IQN-ILS is the MultiVector Quasi-Newton (MVQN) algorithm presented by Bogaers et. al. in [62], which according to the original authors converges slightly faster than the IQN-ILS. This comes however at the price of requiring to compute a square matrix inversion, whose size equals the interface residual one, each time the interface Jacobian needs to be approximated. Such limitation has been not long ago surpassed in [70].

In this work we decide to use a black-box coupling strategy based on QN algorithms. Despite QN methods can be more computationally expensive than JFNK approaches, as they involve a rather large number of linear algebra operations, they are proven to be a robust and efficient approach for the FSI residual minimization. Among all the presented QN algorithms, we select the MVQN method as reference convergence accelerator due to its reported well performance. Besides this, we do not foresee that its matrix inversion becomes in a limitation for the problems we aim to solve in this work.

1.4. Objectives

The main target of this work is to build a robust and efficient FSI framework able to solve strongly coupled problems involving structures that undergo large displacements and rotations, with possible topology changes. Taking these requirements into account, we build an implicit black-box coupling strategy on top of a non-conforming embedded CFD solver.

Besides this, we focus on having a unique framework that works with both thin-walled (shells and membranes) and volumetric structures. This is achieved by switching the level set type depending on the nature of the immersed bodies (i.e. continuous for volumetric bodies and discontinuous for membrane-like ones).

The paper is organized as follows. The 2nd section briefly describes the embedded fluid and structure formulations. The 3rd section details the methodology we follow to achieve the FSI coupling of an embedded CFD solver with a body conforming mechanical one. The 4th section presents our improvement to the FM-ALE method. The 5th section collects the benchmarking of our proposal. A real-life application is also presented in this section. Finally, the conclusions and further work lines are collected in the last section.

All the methods presented in this work are implemented and available to use within the Kratos Multiphysics open source framework [71] [72]. All the models are generated, meshed and post-processed using the GiD pre and post-processor [73][74].

2. Formulation

In this section we describe the formulations that are implemented in each one of the subdomain solvers. We firstly describe the governing equations, stabilization technique and BCs imposition of the embedded CFD solver. Secondly, we briefly present the element technologies we use in the structural solver.

2.1. Fluid domain

2.1.1. Governing equations

In this work we only consider viscous incompressible Newtonian fluids. Therefore, the Cauchy stress tensor $\boldsymbol{\sigma}$ is defined as $\boldsymbol{\sigma} = -p\mathbf{I} + \mathbb{C} : \boldsymbol{\nabla}^s \mathbf{v}$, where $\mathbf{v}$ is the fluid velocity, p the fluid pressure, $\boldsymbol{\nabla}^s$ the symmetric gradient operator and $\mathbb{C}$ the Newtonian constitutive tensor that describes the viscous behaviour.

Inserting the previous definition of $\boldsymbol{\sigma}$ into the linear momentum equilibrium equation and combining the resulting equation with the incompressible (divergence free) mass balance equation yields the well-known viscous incompressible Navier-Stokes (N-S) equations.

Even though this form of the N-S equations is extensively used to solve almost incompressible flows, we opt to introduce a slight compressibility in the mass conservation equation. This is done to avoid the pressure field to be undefined when isolated fluid domains (i.e. bubbles) appear in the level set function. We note that this undesired situation could be immediately dealt with by imposing a Neumann BC inside the isolated fluid cavities. However, it is not always possible to "a priori" known their location, as they likely appear after the level set calculation, either because it is computed from a poorly defined input geometry or due to the deformation of the immersed geometry. In this context, the extra pseudo-compressibility becomes in a numerical tool to ensure that the pressure field remains always bounded.

Hence, we assume that the density ρ can be related to the pressure by the simplified speed of sound c equation of state $p = \rho c^2 \implies c^2 = \partial p / \partial \rho$, which is valid for almost incompressible fluids. By further considering that the density fluctuations are negligible, meaning that $\boldsymbol{\nabla}\rho \approx \mathbf{0}$, we obtain the final form of the pseudo-compressible N-S equations used in this work

$$\rho \frac{\partial \mathbf{v}}{\partial t} + \rho \mathbf{a} \cdot \boldsymbol{\nabla}\mathbf{v} - \boldsymbol{\nabla} \cdot (\mathbb{C} : \boldsymbol{\nabla}^s \mathbf{v}) + \boldsymbol{\nabla} p = \rho \mathbf{b} \tag{1a}$$

$$\frac{1}{\rho c^2} \frac{\partial p}{\partial t} + \boldsymbol{\nabla} \cdot \mathbf{v} = 0 \tag{1b}$$

where $\partial \bullet / \partial t$ and $\boldsymbol{\nabla}$ denote the partial time derivative and gradient operators, $\mathbf{b}$ the volume (body) forces and $\mathbf{a}$ the convective velocity, which in an ALE framework equals $\mathbf{v} - \mathbf{v_m}$ being $\mathbf{v_m}$ the mesh velocity. In fixed mesh approaches, like the one presented used this work, it is commonly assumed that $\mathbf{a} = \mathbf{v}$. However, we highlight that it is essential to keep $\mathbf{a} \neq \mathbf{v}$ for the future application of the FM-ALE algorithm.

We also note that the single-fluid fully-incompressible form $\nabla \cdot \mathbf{u} = 0$ is recovered as $c \to \infty$. Taking this into account, the speed of sound is taken as 10^{12}m/s, so that the compressibility is effectively negligible. Nonetheless, the option to employ a different value is left for those cases in which isolated volumes of fluid may appear.

We refer the reader to [28] for a detailed description of the variational form derivation and implementation of the governing equations.

2.1.2. Stabilization

In this work we only use simplicial elements with same velocity and pressure interpolation order. These elements do to not satisfy the *inf-sup condition*, requiring thus the use of a stabilization method [75]. Among all the techniques that have been proven to be effective in similar CFD problems (e.g. the Finite Increment Calculus [76] or the Galerkin Least Squares (GLS) [77]) we select the Variational Multiscales Method (VMS) [78][79].

The VMS relies on separating the solution fields $\mathbf{v}$ and p in two scales as $\mathbf{v} = \mathbf{v_h} + \mathbf{v_s}$ and $p = p_h + p_s$. $\mathbf{v_h}$ and p_h are the FE resolvable scales while $\mathbf{v_s}$ and p_s are the subscales representing the fluctuations that cannot be captured by the FE mesh and thus need to be modelled. The subscale models are commonly built as a projection of the FE residuals. In our case we apply the most simplest approach, which is to use an identity projection operator, leading to the well-known Algebraic Sub-Grid Scales (ASGS) approach [80].

Hence, the ASGS velocity ($\mathbf{u_s}$) and pressure (p_s) subscales are computed from the FE velocity and pressure solutions as

$$\mathbf{u_s} = \tau_1 \mathbf{R^M}(\mathbf{u_h}, p_h) \tag{2a}$$

$$p_s = \tau_2 R^C(\mathbf{u_h}, p_h) \tag{2b}$$

where τ_1 and τ_2 are the stabilization constants, that we take from [80], and $\mathbf{R^M}$ and R^C are the residuals of the momentum and mass conservation equations, which can be found in [28].

2.1.3. Embedded boundary condition imposition

This subsection presents the two embedded methods that we employ to weakly impose the fluid BCs over the level set intersections. On the one hand, we use the modified Nitsche method presented in [40] for those bodies that have a well defined internal volume, On the other hand, we use the discontinuous formulation described in [46] for the membrane-like bodies that feature no internal volume. In the following, we briefly describe the particularities of each one of these approaches.

Volumetric bodies: Modified Nitsche method no-slip imposition. The modified Nitsche method proposed by Codina and Baiges requires to firstly identify the set of nodes that are in the interior side of the level set, which is to say, the ones that have a negative distance value. Then, the N-S momentum contribution is substituted in these by an L_2 minimization of the velocity constraint error. We refer the reader to [40] for the implementation details and convergence analysis of the method.

We highlight that, despite its simplicity, the method features a remarkably good performance. We also observe that the stability of the method is only compromised when elements with a small portion of fluid appear. This is due to the fact that the interior nodes that enforce the velocity constraint are in this case far from the embedded skin. Therefore, their shape function values are close to zero in the level set intersection, inducing thus the appearance of spurious peak values. As we mention in the introduction of the paper, we tackle these issues with our level set quality check and correction algorithm.

Volumeless bodies: Ausas FE space and Nitsche Navier-slip imposition. Following the ideas presented in [28], we propose to use the Ausas discontinuous FE space (Fig. 5) to represent the elemental velocity and pressure jumps arising from the immersion of a thin-walled body. This capability comes however at

the price of featuring worse interpolation properties than the standard linear FE space as only piecewise constant functions can be exactly represented.

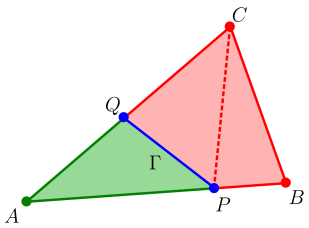
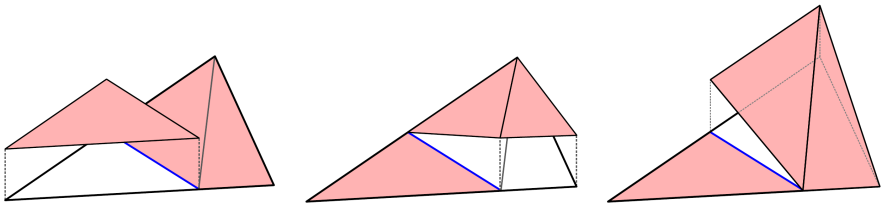


Figure 4: Partition of a triangular finite element ABC into subelements following the interface PQ (source [45]).



(a) N_A (b) N_B (c) N_C

Figure 5: Triangle shape functions for the Ausas finite element space and the splitting pattern in 4 (source [45]).

On top of that, we also follow the ideas presented in [44] to do a Nitsche-based imposition of the Navier-slip condition over the discontinuous level set intersections.

The Navier-slip condition can be understood as a wall law whose behaviour is regulated by the so called slip length parameter ε. Hence, it approaches the no-slip limit as $\varepsilon \to 0$. Contrariwise, it tends to the slip limit as $\varepsilon \to \infty$. Having said this, it is important to remark the capability of the Nitsche method presented [44] to deal with both the slip and the no slip limits without losing its stability.

From a numerical perspective, the Navier-slip condition is indeed a no penetration condition in the normal direction combined with a shear force imposition in the tangential one. This allows to split and express these two contributions as

$$\mathbf{P}^n \left(\mathbf{v} - \mathbf{g}\right) = \mathbf{0} \tag{3a}$$

$$\mathbf{P}^t \left(\varepsilon \left(\left[\mathbb{C} : \left(\boldsymbol{\nabla}^s \mathbf{v}\right)\right] \cdot \mathbf{n} - \mathbf{h}\right) + \mu \left(\mathbf{v} - \mathbf{g}\right)\right) = \mathbf{0} \tag{3b}$$

where $\mathbf{g}$ and $\mathbf{h}$ denote the velocity and the tangential traction to be imposed over the level set intersection. $\mathbf{P}^n$ and $\mathbf{P}^t$ denote the normal and tangential projection operators, which are computed from the unit normal $\mathbf{n}$ as $\mathbf{P}^n = \mathbf{n} \otimes \mathbf{n}$ and $\mathbf{P}^t = \mathbf{I} - \mathbf{n} \otimes \mathbf{n}$, being $\mathbf{I}$ the second order identity tensor.

We note that in the Nitsche imposition of Eq. 3 needs to be applied in both sides of the level set intersections after the ability of the Ausas FE space to disconnect the solutions in each element subdivisions.

More details on the implementation, convergence and accuracy analysis of the discontinuous Nitsche Navier-slip imposition can be found in [46]. This approach is also validated with data from biomedical *in vitro* experiments in [8].

2.2. Structure domain

Bearing in mind that our goal is to solve FSI problems that involve large displacements and (or) rotations, we only consider non-linear finite displacements formulations.

Therefore, volumetric bodies are discretized by using Total Lagrangian linear quadrilateral elements. With regard to Total Lagrangian solid FE, we refer the reader to [81].

Regarding the thin-walled structures, shells are modelled by an enhanced Discrete Kirchhoff Triangular (DKT) based element. This formulation is developed in [82] by combining the DKT plate bending element with the OPtimal Triangle (OPT) membrane element, yielding the so called DKT-OPT element. Such element implements a corotational non-linear formulation capable to efficiently deal with small strain large displacement and rotation problems.

As the benchmarking experiments we reproduce do not require advanced constitutive laws, we can assume throughout the paper that the material response always lays in the elastic regime. Thus, we use the neo-Hookean and Kirchhoff-Saint-Venant material models, which are compatible with the chosen finite displacements formulations.

3. Embedded Fluid-Structure Interaction coupling

In his section we detail the particularities of the embedded FSI coupling. First of all, the embedded DN interface residual is defined together with the non-linear strategy we implement for its minimization. Then, we briefly describe the implementation of such strategy, which is conceived to ease the integration in an existent FSI solver. From now on, we denote the fluid and structure domains as Ω_f and Ω_s while their corresponding FSI interfaces are denoted as Γ_f and Γ_s. Similarly, k superscripts stand for the FSI non-linear iterations counter.

3.1. Dirichlet-Neumann interface residual minimization

Prior to the definition of any interface residual, we recall that the use of an embedded CFD solver entails that Γ_f is implicitly represented by the level set function. As a consequence, the FSI interface residual can only be defined in terms of the structure interface Γ_s.

In this context, the FSI interface residual vector $\mathbf{r}_{\mathbf{\Gamma_s}}^{\mathbf{k+1}}$ to be minimized can be computed from the current and previous iteration displacement results $\tilde{\mathbf{u}}_{\mathbf{\Gamma_s}}^{\mathbf{k+1}}$ and $\mathbf{u}_{\mathbf{\Gamma_s}}^{\mathbf{k}}$ as

$$\mathbf{r}_{\mathbf{\Gamma_s}}^{\mathbf{k+1}} = \tilde{\mathbf{u}}_{\mathbf{\Gamma_s}}^{\mathbf{k+1}} - \mathbf{u}_{\mathbf{\Gamma_s}}^{\mathbf{k}} \tag{4}$$

Here the vector $\tilde{\mathbf{u}}_{\mathbf{\Gamma_s}}^{\mathbf{k+1}}$ is the unrelaxed displacement solution obtained after applying the iteration k+1 fluid load. Conversely, $\mathbf{u}_{\mathbf{\Gamma_s}}^{\mathbf{k+1}}$ is the relaxed displacement obtained after the addition of the solution update $\Delta\mathbf{u}_{\mathbf{\Gamma_s}}{}^{k+1}$ as

$$\mathbf{u}_{\mathbf{\Gamma_s}}^{\mathbf{k+1}} = \mathbf{u}_{\mathbf{\Gamma_s}}^{\mathbf{k}} + \Delta\mathbf{u}_{\mathbf{\Gamma_s}}{}^{k+1} \tag{5}$$

According to our black-box FSI coupling choice, we require the interface residual (Eq. 4) minimization to be done with a non-intrusive method (e.g. relaxation, JFNK or Quasi-Newton). To that purpose we opt for the MVQN method [62], which allows to obtain the correction $\Delta\mathbf{u}_{\mathbf{\Gamma_s}}{}^{k+1}$ from the current iteration interface residual $\mathbf{r}_{\mathbf{\Gamma_s}}^{\mathbf{k+1}}$ and the previous iteration solution $\mathbf{u}_{\mathbf{\Gamma_s}}^{\mathbf{k}}$.

At this point it is important to recall that we tackle the non-linearity of the FSI problem by using a staggered iterative strategy. Hence, we also define an absolute convergence criterion to stop the subdomain iteration when the next residual norm condition is met

$$\frac{\left\|\mathbf{r}_{\mathbf{\Gamma_s}}^{\mathbf{k+1}}\right\|}{\sqrt{n_{\Gamma_s}}} < tol_{FSI} \tag{6}$$

where n_{Γ_s} denotes the number of interface unknowns, whose value equals the structure interface residual $\mathbf{r}_{\mathbf{\Gamma_s}}$ size.

All these ingredients are combined in Algorithm 1, which describes the steps of our embedded FSI iterative strategy. In this algorithm ϕ_f stands for the level set function while the operators $\mathbf{F}$, $\mathbf{P}^{f\to s}$, $\mathbf{S}$ and $\mathbf{CA}$ represent the fluid solver, the interpolator from the fluid background mesh to the structure skin one, the solid solver and the convergence accelerator. We note that all these operators are considered and implemented as independent objects that, from a given input data, give back an output result.

With regard to the pressure interpolation $\mathbf{P}^{f\to s}$, we point out that the same FE space used in the resolution of the CFD problem is employed. Hence, when dealing with volumetric bodies we use the standard linear FE space. On the

contrary, we use the Ausas FE space when volumeless bodies are analysed.

Algorithm 1: Embedded FSI partitioned strategy for DN scheme.

```
k = 0;
max_it = n;
while k ≤ max_it do
    k += 1;
    1) update level set: u_{Γs}^{k} → φ_f^{k+1};
    2) do FM-ALE operations: FMALE(φ_f^{k+1});
    3) solve embedded fluid: F(φ_f^{k+1}) → (v_f^{k+1}, p_f^{k+1});
    4) interpolate pressure to structure skin: P^{f→s}(p_f^{k+1}) → p_{Γs}^{k+1};
    5) solve structure: S(p_{Γs}^{k+1}) → ũ_s^{k+1};
    6) compute residual in Γ_s: r_{Γs}^{k+1} = ũ_{Γs}^{k+1} − u_{Γs}^{k};
    7) update solution in Γ_s: CA(r_{Γs}^{k+1}, u_{Γs}^{k}) → u_{Γs}^{k+1};
    if ‖r_{Γs}^{k+1}‖ / √(n_{Γs}) < tol_FSI then
        break;
    end
end
```

3.2. Implementation: the FSI interface object

In this subsection we describe our Object Oriented (OO) implementation proposal for Algorithm 1. Such implementation approach, which is schematically depicted in Fig. 6, is based on two basic requirements.

First of all, it is mandatory that each subdomain solver (`Fluid Solver` and `Structure Solver` in Fig. 6) works as a stand-alone unit (class) to achieve an effective OO implementation. Secondly, it is important that the communication of the subdomain solvers with the rest of the code is done through the already existent Application Programming Interface (API). The combination of these two requests ensures that there is no need to modify the implementation of neither the fluid nor the structure solvers for the specific purpose of coupling.

The design is completed by the introduction of a `Coupling Interface` object (Fig. 6), which encapsulates the tools used in the coupling, such as, mapping, residual calculation and solution update.

Multiple options exist for the implementation of the `Coupling Interface` object. Our choice is to duplicate it, so to make an independent instance in each single domain interface. In other words, one interface object operates on the $\bar{\Gamma}_f$ interface mesh while the other does so on the $\bar{\Gamma}_s$ one.

In a general case, either of these two coupling interface instances can be used to perform the residual minimization and interface solution update. However, in the embedded case the skin mesh corresponding to the $\bar{\Gamma}_f$ interface object needs to be created as an exact copy of the $\bar{\Gamma}_s$ one. This choice is taken since the Γ_f mesh only exists implicitly, as the zero isosurface of the level set function ϕ_f.

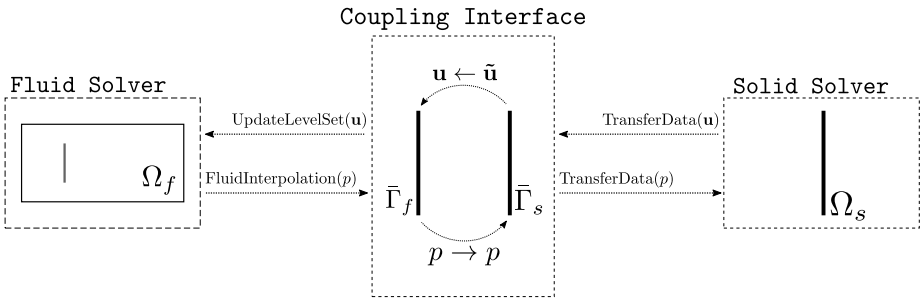


Figure 6: Coupling objects and methods schematic representation. The dashed lines represent the fluid and solid solvers as well as the FSI coupling interface classes. The dotted lines represent the information exchanges.

The DN coupling scheme described in Algorithm 1 can be implemented according to these guidelines. In the following, we tag those operations that are performed in the fluid coupling interface instance with `CI_f`. Equivalently, we mark those ones belonging to the structure interface with `CI_s`. Hence, Algorithm 1 translates to

1. Ω_f: compute the level set function ϕ_f from the $\bar{\Gamma}_f$ interface mesh
2. Ω_f: solve the embedded CFD problem.
3. `CI_f`: interpolate the pressure from Ω_f to $\bar{\Gamma}_f$.
4. `CI_f`→`CI_s`: transfer the pressure from $\bar{\Gamma}_f$ to $\bar{\Gamma}_s$.
5. `CI_s`: transfer the pressure load in $\bar{\Gamma}_s$ to Γ_s.
6. Ω_s: solve the mechanical problem.
7. `CI_s`: get the obtained displacement $\tilde{\mathbf{u}}$ from Γ_s, calculate the interface residual (Eq. 4) and call the convergence accelerator (owned by `CI_s`) to do the solution update $\Delta\mathbf{u}$ (Eq. 5).
8. `CI_s`→`CI_f`: update the $\bar{\Gamma}_f$ interface mesh accordingly to compute the next iteration level set function if convergence is not reached.

We would like to conclude this section highlighting the potential of this implementation. The crucial advantage of this approach is to make the FSI implementation essentially the same for the body-fitted and embedded cases. This can be efficiently achieved by simply overriding some of the auxiliary dotted lines operations in Fig. 6. Furthermore, the `Coupling Interface` can be used out of the box for coupled problems other than the FSI as its API allows customizing the variables the data is retrieved from and sent to.

4. Lagrangian interface FM-ALE algorithm

In this section we target the embedded moving bodies historical data initialization issue described in the introductory section. With that aim, we firstly describe the main features of the FM-ALE algorithm proposed by Codina et.

al. in [48]. Secondly, we present our proposal to compute the FM-ALE mesh moving problem BCs from the motion of the embedded body.

As we have already mentioned, the purpose of the FM-ALE algorithm is to perform a consistent advection of information in the vicinities of the embedded bodies so that their historical values are consistently initialized as their boundaries move across the background mesh.

To this end, a copy of the fixed background mesh $\mathcal{M}^0$ is created at the beginning of the simulation. Such copy is the so called virtual mesh, which we denote as $\mathcal{M}^v$. The idea of the method is to deform $\mathcal{M}^v$, by solving a standard ALE mesh motion problem, to follow the embedded boundaries motion. Then, the historical data, as well as the mesh velocity, are projected from the deformed $\mathcal{M}^v$ to $\mathcal{M}^0$ where the CFD problem is to be solved. Hence, the **FMALE** call in Algorithm 1 can be implemented as described in Algorithm 2. We refer the reader to [48] for a detailed discussion about the implementation and validation of the FM-ALE method.

Algorithm 2: FM-ALE algorithm.

if `virt_mesh` == *false* **then**
  create virtual mesh: $\mathcal{M}^v$ = **CopyMesh**($\mathcal{M}^0$);
  create mesh motion solver: **M**($\mathcal{M}^v$);
  `virt_mesh` = true;
end
1) mesh motion problem;
  1.1) set mesh motion BCs: **M**($\mathcal{M}^v$).**SetBCs**();
  1.2) solve mesh problem: **M**($\mathcal{M}^v$).**Solve**() ;
  1.3) compute mesh velocity: **M**($\mathcal{M}^v$).**ComputeMeshVel**() $\rightarrow \mathbf{v_m}$;
2) data interpolation from $\mathcal{M}^v$ to $\mathcal{M}^0$;
forall $\texttt{node}_\texttt{i} \in \mathcal{M}^0$.Nodes() **do**
  2.1) find $\texttt{node}_\texttt{i}$ in $\mathcal{M}^v$.Elements() $\rightarrow \texttt{elem}^\texttt{v}$;
  2.2) interpolate $\mathbf{v_i^n}$, $\mathbf{v_i^{n-1}}$, $\mathbf{v_{m_i}}$, p_i^n, p_i^{n-1} within $\texttt{elem}^\texttt{v}$;
end
3) revert $\mathcal{M}^v$ movement;

At this point, it is interesting to discuss which are most suitable BCs for the mesh motion problem in the step 1.1 of Algorithm 2. Taking the body fitted case as reference, it is required that the FSI interface has a Lagrangian motion to ensure the mass conservation. This turns into enforcing $\mathbf{u}_m^\Gamma = \mathbf{u_s^\Gamma}$ and $\mathbf{v_m^\Gamma} = \mathbf{v_s^\Gamma}$. However, this cannot be imposed in a strong sense when using an embedded discretization due to the implicit representation of Γ_s.

With this regard, we present a slight improvement to the FM-ALE algorithm to approximatively fulfil such Lagrangian motion of the interface. The proposal is based on modifying the step 1.1 in Algorithm 2 to impose a set of $\mathbf{u_m}$ nodal values in the split elements such that their interpolation in the level set intersections yield $\mathbf{u_s^\Gamma}$. This ensures that the mesh displacement field obtained after the deformation of $\mathcal{M}^v$ (step 1.2 in Algorithm 2) matches the movement of Γ_s in an approximate Lagrangian sense.

In Fig. 7a we present a simple toy example to explain our proposal. In this figure, the background mesh is represented with solid black lines while the intersecting structure skin is represented with dashed red lines. To ease the discussion we assume that the variable to extrapolate, which we denote as a^{Γ}, is scalar. Nonetheless, the same procedure can be identically applied to a vector variable by extending what is explained in here to each one of the vector components. We note that in the FM-ALE case, the variable of interest is the structure displacement increment $\mathbf{\Delta u_s} = \mathbf{u_s^{n+1}} - \mathbf{u_s^n}$.

The proposed technique consists in solving an extrapolation problem on an auxiliary mesh made up with the background mesh intersected edges. This problem is implemented within the step 1.1 of the FM-ALE algorithm 2 by adding the following sub-steps

1. find those elements that are intersected by the level set. For each intersected element, isolate their intersected edges (Fig. 7b).
2. for each intersected edge e_{ij}, obtain the point $\mathbf{x^{int}}$ (light green square dots in Fig. 7b) in where the edge intersects the structure skin. Save the distance $d_e = \left\| \mathbf{x^{int}} - \mathbf{x^0} \right\| / l_e$, where $\mathbf{x^0}$ and l_e are the i-node coordinates and length of e_{ij}.
3. interpolate and save the value of the variable of interest a^{Γ} from the structure skin to the intersection points $\mathbf{x^{int}}$.
4. create one extrapolation line element from each intersected edge
5. solve the problem in the green light nodes auxiliary mesh (Fig. 7c).

After the resolution of the edge mesh extrapolation problem, we obtain a set of nodal values $\mathbf{a_{ij}}$ whose interpolation in the level set intersections (light green square dots in Fig. 7c) equals the a_{Γ} values previously obtained from the structure skin Γ_s. Besides the extrapolation contribution, we add an extra term that penalizes the gradient of the solution values $\mathbf{a_{ij}}$. This prevents the appearance of spurious values in those situations in which the solution is not unique (e.g. isolated edges).

Hence, for an intersected edge e_{ij} the problem takes the form

$$\mathbf{LHS_{ij}} \mathbf{a_{ij}} = \mathbf{RHS_{ij}} \tag{7}$$

where the left and right hand sides of the problem are defined as

$$\mathbf{LHS_{ij}} = \begin{pmatrix} N_i N_i & N_i N_j \\ N_j N_i & N_j N_j \end{pmatrix} + \begin{pmatrix} \kappa^G \kappa^G & -\kappa^G \kappa^G \\ -\kappa^G \kappa^G & \kappa^G \kappa^G \end{pmatrix} \tag{8}$$

and

$$\mathbf{RHS_{ij}} = \begin{pmatrix} N_i a^{\Gamma} \\ N_j a^{\Gamma} \end{pmatrix} \tag{9}$$

N_i and N_j are computed from the edge intersection distance d_e as $N_i = 1 - d_e$ and $N_j = d_e$ while κ^G is the solution gradient penalty constant. According to our experience, small values of κ^G (of the order of 10^{-3}-10^{-4}) are sufficient to drive the problem towards a smooth solution.

We would like to conclude this section saying that taking the extrapolation problem values as mesh motion BCs guarantees that the movement is large enough to properly initialize the historical values but sufficiently small to avoid the common mesh motion problems (i.e. element distortion and inversion). Last but not least, we also highlight that the extra computational effort of the extrapolation problem is negligible compared to the mesh motion one as it is exclusively solved in those nodes belonging to the intersected edges.

5. Validation

In this section we present the validation of the proposed embedded FSI framework. We base the correctness assessment on the comparison with reference data available in the literature. Whenever it is possible, we also compare the embedded solution with the ones obtained with our reference body fitted FSI solver.

Bearing in mind that one of the main motivations for the use of an embedded FSI solver is to get rid of the ALE mesh motion problems, we only consider examples that could be challenging from the mesh motion perspective. Therefore, all the problems presented in this section involve structures that undergo large displacements and (or) rotations.

Besides this, embedded mesh methods can be also exploited to efficiently simulate membrane-like structures, allowing to circumvent all the meshing complexities (i.e. interface entities duplication) associated to the volume meshing of structures without internal volume. In this regard, we also present cases that imply the analysis of thin-walled bodies.

Taking this into consideration, on the one hand we solve three examples involving volumetric bodies. The first two are the well-known Mok and the Turek & Hron benchmarks. The third one is proposed by us to prove the capabilities of embedded mesh methods when dealing with arbitrary large rotations. It consists in a two-dimensional mixer with flexible blades. On the other hand, we also present two three-dimensional cases that involve thin-walled structures. The first one is a simple lid-driven cavity problem with a flexible membrane in the bottom. The second one is a potential industrial application of the presented FSI framework that is the FSI analysis of a four-point tent built as a system of beams and cables supporting a prestressed membrane.

5.1. 2D Mok benchmark

5.1.1. Problem description

The Mok benchmark was firstly presented in [65]. It consists in a convergent channel with an internal flexible wall. The main challenge of the problem is that the fluid and structure densities have a similar order of magnitude, leading to a strongly coupled problem with large interaction between the two fields.

The geometry and boundary conditions are depicted in Fig. 8. As it can be observed, the top edge is set as a slip boundary while the inferior one is no-slip.

The pressure is imposed to zero in the right edge. A parabolic inlet defined as

$$v_x(y,t) = 4\bar{v}y(1-y) \ \ [m/s] \tag{10}$$

is applied to the left edge. $\bar{v}$ is a time dependent reference velocity such that

$$\bar{v} = \begin{cases} \frac{0.06067}{2}\left(1-\cos\frac{\pi t}{10}\right) & if\ t \leq 10 \\ 0.06067 & otherwise \end{cases} \tag{11}$$

The fluid and structure material properties are collected in Table 1. We highlight that the structural material response is modelled using a unit thickness plane stress linear elastic constitutive law. The simulation runs from 0s to 25s with a fixed time step of 0.1s. The FSI coupling tolerance is set to 10^{-8}.

Table 1: 2D Mok benchmark. Material properties.

Structure		Fluid	
ρ_s	1.5e3 kg/m^3	ρ_f	956.0 kg/m^3
E	2.3e6 N/m^2	μ	0.145 $Pa \cdot s$
ν	0.45		

As it is done in the original work by Mok [65], as well as in [83] by Valdés, we take the structure control points A and B u_x displacement evolution as comparison magnitude. These two reference nodes are located at the top and centre points of the windward edge of the wall (Fig. 8).

Some discrepancy can be observed with respect to the u_x results reported in [65] and [83]. As the mesh used in the current examples is however much finer than the one employed in the two references, we decided to perform a preliminary body-fitted mesh convergence study to obtain a comparable reference solution. Therefore, we solve the Mok benchmark problem with three different body fitted meshes: mesh 0, mesh 1 and mesh 2. Mesh 0 is generated as similar as possible to the one described in [83]. Mesh 1 is created by halving the element sizes of mesh 0. Similarly, mesh 2 is created by doing so with the element sizes of mesh 1.

Table 2 collects the number of structure and fluid elements ($n_{el}^{\Omega_s}$ and $n_{el}^{\Omega_f}$) employed in each case together with the maximum u_x) obtained in both control points A and B. Furthermore, Fig. 9 compares our u_x results with the ones in [65] and [83]. We observe that the solution converges to larger values, specially when the peak displacement occurs, than the ones reported in [65] and [83].

Accordingly, from now on we consider as reference solution the one obtained with our body fitted solver and the finest mesh discretization level (mesh 2).

With these results in mind, and for the sake of a fair comparison, we set our embedded mesh as similar as possible to the finest body fitted one (mesh 2). Hence, the background fluid mesh (Fig. 10) is formed by 82k linear triangular

Table 2: 2D Mok benchmark. Body fitted convergence study.

	Mok [65]	Valdes [83]	Body fitted		
			Mesh 0	Mesh 1	Mesh 2
$n_{el}^{\Omega_s}$	6 (quadratic)	6 (quadratic)	0.4k	1.6k	6.4k
$n_{el}^{\Omega_f}$	1.9k	6k	6.6k	18.8k	65.5k
$max(u_x^A)$ [cm]	7.789	8.190	8.363	8.756	8.856
$max(u_x^B)$ [cm]	2.743	2.917	3.052	3.204	3.244

elements (Q1P1)[2].

As it is done in the body fitted case, the mesh is refined in the region close to the structure in order to properly represent the embedded coupling interface (Fig. 10b). However, we note that in the embedded case it is not possible to "a priori" know the location of the structure interface. As a consequence, the mesh refinement needs to be done in the region in where the embedded structure is expected to move across, rather than only refining the geometry edges of interest. This yields thus a slightly larger number of elements for the same refinement level.

Finally, we use the same structure mesh employed in the finest body fitted case (mesh 2). Hence, the embedded structure mesh is also meshed with a total of 6.4k linear quadrilateral Total Lagrangian elements.

5.1.2. Results assessment

The Figs. 11 and 12 present some snapshots of the v_x and p solution fields together with the corresponding zero isosurface of the distance function. As expected, the structure undergoes a large displacement, whose value is maximum at the end of the first transition phase. Once the inlet velocity becomes constant in time, the solution reaches a steady state, whose maximum deflection value is around 7cm.

Taking into account that this example involves a low Re number flow, we decide to solve the reference body fitted case (mesh 2) with and without considering the wall shear component in the structure load. The obtained solutions are compared in Figs. 13 and 14. As it can be observed, the embedded u_x time evolution in both control points is in very good agreement with the body fitted reference ones (Fig. 13a and 14a). Nonetheless, we note minor differences in the peak displacement values (Figs. 13b and 14b). We also notice that such differences are almost imperceptible when the embedded solution is compared to the body fitted one without the wall shear component (denoted as "Body fitted (mesh 2) - p" in Figs. 13 and 14).

On top of the wall shear stress contribution, we believe that the small de-

[2]Henceforth, we follow the same element nomenclature as in [14]. Therefore, Q1P1 states for a linear velocity and pressure interpolation element.

viations in the results are possibly due to the intrinsic particularities of the embedded CFD formulation, which we recall inherently smooths sharp corners as these cannot be represented in terms of a continuous distance function. Considering that the control point A coincides with one vertex of the structure, minor discrepancies in the values can be expected.

With regard to the computational efficiency, we also compare the number of required FSI coupling iterations. The results depicted in Fig. 15 show that the embedded solver is slightly more efficient than the body fitted one. We think that these results cannot be taken as a general rule. Instead, what we can asseverate is that the use of an embedded framework in the fluid domain does not deteriorate the convergence of the FSI coupling.

Finally, we would like to point out that this problem can be alternatively, and possibly more efficiently, solved by modelling the flexible wall as a non-linear beam and using discontinuous level set based formulation in the fluid.

5.2. 2D Turek & Hron benchmark - FSI2

5.2.1. Problem description

The second example we present is the well-known Turek & Hron benchmark (also known as Turek benchmark). In their original work [84], Turek and Hron present three FSI variants, being the second and the third ones the most challenging ones. Since we want to exploit the capabilities of embedded formulations to deal with large boundary displacements, we choose the second variant, denoted by the original authors as FSI2. In this case the problem settings are selected in such a way that the structure undergoes extremely large deflections, being thus a tough example from the mesh motion perspective.

The problem geometry consists in a rigid cylinder inside a straight channel (Fig. 16a). Attached to the cylinder, there is a flexible cantilever beam (Fig. 16b). The problem geometry is depicted in Fig. 16 and its dimensions are H 0.41m, L 2.5m, h 0.02m, l 0.35m and r 0.05m. The centre of the cylinder is placed at (0.2,0.2)m while the coordinates of control points A (initial position) and B are (0.6,0.2)m and (0.15,0.2)m.

Concerning the boundary conditions, the pressure is fixed to zero along the right edge of the channel. The velocity is fixed to zero in both top and bottom edges as well as in the cylinder boundary. In the left edge, the parabolic inlet function

$$v_x(y) = 1.5\bar{v}\frac{y(H-y)}{\left(\frac{H}{2}\right)^2} = 1.5\bar{v}\frac{4}{0.1681}y(0.41-y) \;\; [m/s] \tag{12}$$

is applied. $\bar{v}$ is a reference velocity that equals 1m/s in the FSI2 case. The previous inlet function is multiplied by an initial ramp-up function to obtain the final velocity profile

$$v_x(y,t) = \begin{cases} v_x(y)\frac{1-cos\left(\frac{\pi}{2}t\right)}{2} & if\ t \leq 2 \\ v_x(y) & otherwise \end{cases} \tag{13}$$

As is done all throughout the paper, the fluid material response is modelled using a Newtonian constitutive law. Conversely, a Kirchhoff - Saint-Venant material model is used in the structure. The material properties of both the fluid and the structure are collected in Table 3. The problem is run for 20s using a time step of 2e-3s, which equals the one used in [84]. The non-linear FSI tolerance is set to 10^{-8}.

Finally, the computational domain is meshed using linear triangular elements (Q1P1) in the fluid domain and linear quadrilateral Total Lagrangian elements in structural one. As in this case we aim to compare the embedded solution to the one obtained with a body fitted alternative solver, the background fluid mesh is set as similar as possible to the body fitted one. Therefore, the number of elements of the embedded background and body fitted fluid meshes is 63.5k and 28k, which we consider reasonably close taking into account that elements that fall within the embedded structural domain are deactivated. Besides this, we also note that, as in the previous example, the differences in the total number of elements are due to the local refinement we do in the leeward side of the cylinder. By doing this we ensure that the level set function is computed with an accuracy level somewhat comparable to the body fitted discretization. With regard to the structure domain, the same structured mesh made up with 2.6k elements is used in both the embedded and the body fitted simulations.

Table 3: 2D Turek & Hron benchmark - FSI2. Material properties.

Structure		Fluid	
ρ_s	10e3 kg/m^3	ρ_f	1000 kg/m^3
μ_s	0.5e6 kg/ms^2	μ_f	1 $Pa \cdot s$
ν	0.4		

5.2.2. Results assessment

Figs. 17 and 18 collect snapshots of the obtained embedded velocity and pressure fields together with the level set zero isosurface representing the structure skin. Because the solution is periodic in this case, we present these magnitudes for a complete period of oscillation (T) starting at a reference time t_0. It can be observed that the periodic vortex shedding generates a pressure pattern that yields the expected large deflections in the cantilever beam.

The structure deflection evolution is compared with the body fitted solver one in Figs. 19 and 20. Even though the transition phase is shorter in the embedded case, barely differences between both solutions can be observed once the periodic regime is reached. Complementary, Figs. 19b and 20b zoom in on the embedded and body fitted solutions from time 15s to 16s, confirming that the peak values and oscillation periods are almost identical in both solutions.

Table 4 collects the mean and amplitude values of the displacement for the last period of oscillation. These values are in good agreement with the reference ones reported in [84], being the embedded solution values slightly larger.

Finally, we would like to add that for this test case the embedded solver, is remarkably robust. On the contrary, it tends to be challenging to achieve a similar robustness when using the body fitted approach, mainly because of the mesh deformation solver, which tends to yield excessively distorted (or even inverted) elements due to the large deformations and rotations of the tail. Conversely, the embedded solver, which worked out of the box, bypasses all these mesh motion issues.

Table 4: 2D Turek & Hron benchmark - FSI2. Point A displacement values [m].

	$u_x \ [\times 10^{-3}]$	$u_y \ [\times 10^{-3}]$
Body fitted	-14.76 ± 12.64	1.29 ± 81.21
Embedded	-15.29 ± 12.77	1.31 ± 83.15
Turek & Hron [84]	-14.58 ± 12.44	1.23 ± 80.60

5.3. 2D mixer with flexible blades

5.3.1. Problem description

So far we have presented two examples that are considered as reference benchmarking test cases in the FSI community. Although they served us to prove the accuracy of our embedded proposal, they do not highlight the specific capabilities of the new method since they can also be solved by employing ALE approaches. In order to prove the extended scope of application of the proposed technique, we present next an example that involves extremely large rotations, and that would be thus impossible to solve by using body fitted approaches.

The problem is conceived as a 2D idealization of a turbine mixer with clockwise-anticlockwise alternate rotation. Hence, the fluid geometry consists in a unit length square domain whose centre coincides with the coordinates origin. The structure geometry represents the three blades of the mixer. These are also centred in the coordinates origin and aligned such that there is a 120°radial symmetry. The blades are 0.375m long (measured from the origin) while their thickness is 0.05m.

With regard to the BCs, a slip BC is enforced in the outer walls while a no-slip behaviour is assumed in the embedded FSI coupling interface. Besides this, we also fix the pressure to zero in the entire wall to obtain a radially symmetric solution. Concerning the structure domain, a rigid body rotation with angular velocity ω equal to $\pm$1rad/s is imposed in the centre of the blades (pink region in Fig. 21b) to emulate the spinning of the mixer, which changes its direction after each lap (2π rad) is completed.

The material properties are collected in Table 5. We note that the stiffness of the blades is intentionally set to an unrealistic value. Although this value might be unfeasible when comparing with a real mixer, it allows to test our solver under the action of not only large rotations but also displacements. On top of that, we also set $\rho_f = \rho_s$ to yield a strongly coupled behaviour. As in

the previous example, the material response is modelled using a Newtonian and Kirchhoff - Saint-Venant constitutive models.

Table 5: 2D mixer with flexible blades. Material properties.

Structure		Fluid	
ρ_s	1.0 kg/m^3	ρ_f	1.0 kg/m^3
E	1.0e2 N/m^2	μ	$1.0e-2$ $Pa \cdot s$
ν	0.3		

The problem is run for 20s to reproduce three complete rotations (anticlockwise, clockwise and anticlockwise) of the mixer. The time step is 0.01s while the non-linear FSI tolerance is 10^{-7}.

Finally, both the fluid and the structure domains are meshed using structured meshes. On the one hand, we use a structured mesh with 110 perimeter subdivisions in the fluid domain. This amounts a total of 24,2k linear triangular elements (Q1P1). On the other hand, each one of the blades of the rotor is meshed with a 8x39 grid made with linear quadrilateral Total Lagrangian elements. In the rigid body motion region of the rotor we use 64 triangular elements to enforce the rotation.

5.3.2. Results assessment

Prior to any results assessment, we note that what we present in the following is a qualitative analysis of the obtained solution due to the lack of reference data.

Hence, we firstly focus on the displacement evolution of one of the three blades of the mixer. To this purpose, we present in Fig. 22 the time evolution of one control point in the blade that is initially in a vertical position. The initial coordinates of such control point are (0,0.375)m and coincide with the short edge centre point of the blade. We note that as the problem has radial symmetry, there is no need to assess the three blades as they behave exactly the same.

From the results in Fig. 22, we observe that after each n rotation is completed (values at time $n2\pi$) the tip of the blade almost recovers its original position. This can be also observed in Fig. 23 which tracks the evolution of the control point coordinates.

To better comprehend the deformation of the flexible blades, we present in Fig. 24 a collection of snapshots that depict the structure during the second rotation (ω -1rad/s and $2\pi \leq t \leq 4\pi$). In Fig. 24a it can be observed that at the end of the first anticlockwise rotation (t $\approx 2\pi$ s) the structure almost recovers the initial spatial configuration. Then, the rotation direction is suddenly inverted, thus inducing a high acceleration in the structure. As a consequence of its small stiffness, such acceleration yields a large deformation in the blades, (Figs. 24b and 24c) that is recovered as the rotation evolves.

This transient-to-steady cycle is repeated at each rotation and can be also observed in the fluid domain. As it can be noted in Figs. 25 and 26, at the

end of each rotation the flow reaches a steady solution, which presents a radial velocity pattern that grows from zero to a maximum value around 0.4m/s in the ends of the blades (Fig. 25a). However, when the rotation changes its direction the flow tends to accordingly do so. Hence, the sudden inversion of the rotation generates three initial pressure (Fig. 26b) and velocity (Fig. 25b) peaks in the three corresponding ends of the flexible blades.

Last but not least, we report the history of FSI non-linear coupling iterations required to reach convergence in Fig. 27. Although the number of FSI iterations varies as the problem evolves, it remains always bounded between 4 and 8 iterations, being in most cases 6 or 7 iterations. This is a remarkable result considering that the densities of the fluid and structure are the same.

Once arrived to this point, we think that the obtained solution feasibly represents the physics of the experimental set up, proving again the correctness of the solver. On top of that, we highlight the capability of our proposal to efficiently solve not only large displacement problems, but also arbitrary extremely large rotation ones. Considering that to solve this problem with a traditional ALE-based solver would definitively require remeshing, we dare say that this is a proof of the enhanced robustness that embedded approaches have to face such extremely large rotation scenarios.

5.4. 3D lid-driven cavity with flexible shell

5.4.1. Problem description

The most distinguishable feature of the example we present in this section is that it involves a thin-walled structure. Considering that one of the advantages of embedded mesh methods is the capability to efficiently deal with complex volumeless bodies, this example aims to test the performance of the proposed FSI solver when these structural typologies are analysed. On top of that, this example also tests the implementation in a 3D scenario.

The problem geometry is taken from the cavity with a thin bottom shell benchmark presented in [65] and consists in a 1x1x1m fluid cavity (Fig. 28a) whose bottom face is a 2.0e-3m thickness flexible membrane (Fig. 28b).

Assuming that the bottom flexible membrane is placed in the 0m z-coordinate, the oscillatory velocity profile

$$v_x = \frac{z - 0.875}{0.125} \left(1 - cos\left(0.4\pi t\right)\right) \ [m/s] \tag{14}$$

is imposed in the surfaces highlighted in Fig. 28c (the v_y and v_z components are fixed to zero). The pressure is fixed to zero in the surface depicted in Fig. 28d. A no-slip BC is set in the remaining lateral faces.

Concerning the constitutive behaviour, a Newtonian model is used in the fluid domain while an elastic one is used in the shell structure, which has no initial prestress. The corresponding material properties are collected in Table 6.

The same fluid and structure meshes are employed in the body fitted and embedded simulation. The fluid volume mesh (Fig. 28a) has around 117k linear

tetrahedra elements (Q1P1) while the structure one (Fig. 28b) is made with 5.5k DKT-OPT shell elements [85]. The problem is run for 60s with a time step equal to 2.5e-2s. The FSI coupling tolerance is set to 10^{-6}.

Table 6: 3D lid-driven cavity with flexible shell. Material properties.

Structure		Fluid	
ρ_s	500.0 kg/m^3	ρ_f	1.0 kg/m^3
E	250.0 N/m^2	μ	0.01 $Pa \cdot s$
ν	0.0		

5.4.2. Results assessment

Fig. 29 depicts the vertical displacement evolution (u_z) of the shell midpoint. As expected, the deformation becomes periodic after a first transition phase. The obtained solution is compared with the one in [83] and also with the one obtained with the reference body fitted solver. We observe that both solutions are similar, in amplitude and period of oscillation, to the one reported by Valdés in [83]. However, there are noticeable differences in the peak oscillation values. Taking into account the good agreement between the embedded and body fitted solutions, we believe that such differences come from the mesh discretization, which is much finer in our case.

Furthermore, Table 7 compares the values that characterize the oscillatory regime. We highlight the remarkably good correlation of the oscillation period (T) and minimum u_z values between the embedded and body fitted solutions. Nonetheless, there is a noticeable difference in the maximum u_z value. This is explained in Fig. 30, which compares the body fitted and embedded pressure fields when the maximum structure deflection occurs. As it can be observed, the mass below the bottom membrane is considered in the embedded problem (Fig. 30b). On the contrary, this is not taken into account in the body fitted case (Fig. 30a). The movement of such extra mass turns into a vertical overpressure, whose maximum value appears when the bottom shell undergoes the maximum deflection, that explains the differences in the maximum u_z values.

Finally, we present two snapshots of the embedded shell displacement norm $\|\mathbf{u}\|$ during the periodic oscillation regime (Fig. 31). The first one (Fig. 31a) depicts the deformed geometry of the shell for the minimum u_z periodic displacement. Conversely, the second one (Fig. 31b) does so for the maximum u_z periodic displacement.

5.5. 3D four-point tent

5.5.1. Industrial motivation

This last example presents a feasible industrial application of the proposed embedded FSI tool. The problem at hand is the FSI analysis of a four-point tent under the action of an intense wind load.

Table 7: 3D lid-driven cavity with flexible shell. u_z oscillation magnitudes.

	$\min(u_z)$ $[m]$	$\max(u_z)$ $[m]$	A $[m]$	T $[s]$
Valdés [83]	0.163	0.251	0.088	5.1
Body fitted	0.192	0.276	0.084	4.8
Embedded	0.189	0.287	0.098	4.75

Since their appearance in the mid past century, there has been an always growing interest on this kind of highly flexible structures. Although they were initially conceived as more appealing alternatives to traditional structural typologies, their versatility and efficient manufacturing encouraged designers to use them as reference structural solutions in some specific applications [86]. Some examples are the design of extremely lightweight roof covers, such as the Olympiastadion München one by Frei Otto, or the design of sustainable and versatile temporary structures (e.g. inflatable hangars or rapid deployment emergency shelters) [87].

In this context, the use of the embedded approach circumvents all the preprocessing issues that appear in the membrane boundaries during the fluid volume mesh generation. On top of that, these structures are known to suffer from large displacements and rotations, which might even turn into topology changes if self-contact and wrinkling appear. In this regard, the embedded approach is a robust alternative to efficiently handle this situation.

5.5.2. Problem description

The geometry of the problem consists in a four-point tent inside a straight wind channel. Fig. 32 collects multiple views of the structure geometry. As it can be observed, the four-point tent is made as a system of beams and prestressed cables supporting a prestressed membrane, whose edges are prestressed cables too (Fig. 33). The membrane is a 1e-3m thickness hyperbolic paraboloid, whose plan size and elevation are 8x8m and 4m.

The wind channel geometry is a 100x40x20m rectangular prism (Fig. 34), which yields a blockage coefficient of the 4%. The membrane centre is located in the point (4,4,0)m of the channel, which is 24m far from the inlet yz-plane.

The fluid BCs are rather simple in this case. A constant x-velocity of 50 m/s is imposed in the inlet region (Fig. 34a) while the other two components are set to zero. The pressure is fixed to zero in the outlet region (Fig. 34b). Moreover, we enforce a slip (symmetry) condition in rest of the channel walls (Fig. 34c). In the embedded FSI interface, we set the slip length ε to 10^{-3}.

We remark that the slip BC is also applied to the ground wall. Although the slip BC is only well defined for inviscid fluids, it is a good approximation in high Reynolds number (Re) scenarios [88]. Bear in mind that imposing a no-slip condition in the ground would have required the use of an overkill boundary layer mesh to properly capture the wall viscous effects.

Concerning the structural problem BCs, the displacements are constrained

in the ground, that is to say in the bottom ends of the supporting beams and cables as well as in two of the membrane corners (Fig. 33d).

The fluid and structure material properties are collected in Table 8. For the cables (Fig. 33b) we present the density ρ_c, the cross section area A_c, the Young modulus E_c and the prestress σ_c^0. The same information is provided for the supporting beams (Fig. 33a) in ρ_b, A_b, E_b and σ_b^0. Equivalently, ρ_m, ν_m, E_m and $\boldsymbol{\sigma_m^0}$ stand for the prestressed membrane density, Poisson ratio, Young modulus and x,y and z-directions prestress. ρ_f and μ_f are the fluid density and dynamic viscosity. An elastic material response is used in all the elements that conform the structure while a Newtonian one is used in the fluid.

Both the supporting beams and cables are meshed with a single element. An average element size of 0.15m is used in the prestressed membrane as well as in its edge cables. These mesh settings yield a 21k elements mesh. For the beams we use standard non-linear Total Lagrangian elements while we use standard non-linear Total Lagrangian truss ones for the cables [81]. The prestressed membrane element formulation is taken from [89].

As it can be observed in Fig. 34 the fluid volume mesh is divided in three refinement regions. The 1st region (Fig. 34d) is the one surrounding the four-point tent and has an average element size of 0.15m. The 2nd one (Fig. 34e) has 0.5m (in the vicinities of the 1st region) and 1m as average element sizes. The 3rd region (the outer one) has an average element size of 2.5 m. Altogether, the three refinement levels amount a total of 3M linear tetrahedra Q1P1 elements.

Table 8: 3D four-point tent. Material properties.

Structure						Fluid	
Cables		Beams		Membrane			
ρ_c	8300.0 kg/m^3	ρ_b	8300.0 kg/m^3	ρ_m	800.0 kg/m^3	ρ_f	1.225 kg/m^3
A_c	$1.5e-4$ m^2	A_b	$7.06858e-2$ m^2	ν_m	0.2	μ_f	$1.846e-5$ $Pa \cdot s$
E_c	200.0e9 N/m^2	E_b	200.0e9 N/m^2	E_m	300.0e3 N/m^2		
σ_c^0	1333.3e6 N/m^2	σ_b^0	0.0 N/m^2	$\boldsymbol{\sigma_m^0}$	$[5.0e6, 5.0e6, 0]$ N/m^2		

5.5.3. Results assessment

Owing to the lack of experimental results, what we present in the following is therefore a description and qualitative assessment of the obtained solution.

First of all, we show in Fig. 35 the time evolution of the three displacement components of a control node whose initial xy-coordinates are (5.25,3.25)m. We select this point because it is located in the region where the structure undergoes the maximum displacements. As it can be observed, all the displacement components show an almost periodic behaviour after an initial transition regime. We also note that the displacement in the wind direction (x-component) is around 0.2m while the vertical one (z-component) is around 1.0m.

Furthermore, Fig. 36 presents some snapshots of the deformed geometry together with the corresponding displacement norm field $\|\mathbf{u}\|$. This figure also compares the deformed geometry with the initial configuration (Fig. 36c).

Besides that, the Fig. 37 collects a set of cross sections of the fluid pressure field p. All these sections are parallel to the xz-plane, which is aligned with the wind direction. Likewise, the Fig. 38 depicts the corresponding pressure interpolation on both sides of the membrane structure.

As it is expected, a flow separation occurs in the front region of the tent. This generates an upwards overpressure (Fig. 38) that induces the positive displacements in the vertical direction (Fig. 36c). On the contrary, a downwards overpressure appears in the rear part of the tent. Such pressure load in the negative direction is indeed much more relevant than the positive one as the maximum deflections are located in this area of the membrane (Figs. 36b and 36c).

Although we are confident about the accuracy and reliability of the obtained results, these also reveal some of the possible further enhancements. On the one hand, we notice some issues with the pressure interpolation in the vicinities of the membrane boundaries (Fig. 38). More specifically, these can be easily observed by focusing on the two top membrane edges in Fig. 38a and on the right one in Fig. 38b. These phenomena, which are also reported in [8], are due to the fact that we do not consider any specific treatment of the edge intersected elements (i.e. the intersected elements neighbouring with non-intersected ones). Thus, we can say that this becomes in a "variational crime" that propagates along the contour of the membrane yielding such edge pressure oscillations.

On the other hand, we note that the mesh resolution in the membrane surroundings is not as good as the one that could be achieved by using an equivalent body fitted mesh (Fig. 37). Nonetheless, a reasonably similar resolution can be always achieved upon mesh refining.

Summarizing, the test showcases the capabilities of the method for a realistic application. Aside of the simplifications during the model generation, which allow to bypass the volume meshing of a thin-walled structure, the embedded solver is proved to efficiently solve the problem out of the box. We also note that this example could be a tough case for a traditional ALE mesh motion based solver. Last but not least, we highlight the relevance that these results have for the possible extension of the presented tool to industrial applications.

6. Conclusion

In this work we present an embedded framework for the resolution of strongly coupled FSI problems involving structures of any type. On top of the idea that any structure can be classified as volumetric or volumeless, we put special emphasis on creating a single FSI tool to deal with both types of bodies.

Besides this, the paper also includes some technical advances that could be eventually used in other contexts. These are

- the use of a discontinuous FE embedded approach for the CFD analysis of volumeless structures. This approach is based on the combination of the Ausas discontinuous FE space with a Nitsche imposition of the Navier-slip wall condition.

- the improvement of the FM-ALE algorithm to consider a Lagrangian motion of the embedded interface. Our proposal is to solve a small extrapolation problem in the intersected edges to find the BCs of the FM-ALE mesh motion problem. Such extrapolation problem is built so that the interpolation of its results over the embedded skin equals (in a variational sense) the structure mesh displacement.

- the black-box FSI OO implementation. We put special interest on the implementation of our strongly coupled staggered strategy. For the sake of modularity and code reusability we use a black-box coupling approach with a Quasi-Newton interface residual minimization. We highlight the flexibility of the OO implementation we follow, which makes possible to solve not only FSI problems but also alternative multiphysics ones (e.g. Conjugate Heat Transfer) or to easily integrate it in already existent codes.

We validate our proposal by solving several examples involving both volumetric and membrane-like bodies. The first two examples are the well-known Mok and Turek & Hron benchmarks. The results of the embedded formulation are in really good agreement with the reference ones, as only minor discrepancies, which we believe are associated to the differences in the discretization as well as in the CFD formulation, can be observed. We also highlight the robustness of the embedded approach, which worked out of the box in both cases.

The third example (two-dimensional mixer with flexible blades) is specifically conceived to prove the robustness of the embedded approach to deal with arbitrary and extremely large rotations. Despite the lack of reference results, we observe that the obtained solution is perfectly feasible. We highlight the value of our method to deal with such a challenging scenario, which would have definitively required remeshing if an ALE-based solver was used.

With regard to thin-walled structures, we present two examples involving such structural typologies. The first one is a lid driven cavity problem with a flexible shell structure in the bottom. Although we observe that the embedded deflections are slightly larger than the reference ones, the period and amplitude of the oscillation are however almost identical to the reference ones.

The second one is the the FSI analysis of a 4-point tent during a extreme wind load episode. We note that, aside of further assessing the performance of the solver when volumeless bodies are considered, this problem represents a feasible real-life application of the proposed technology. Despite the lack of reference results, we can say that the solver accurately solves the problem and is capable of capturing the flow features that induce the large deflection of the membrane. These results make us be confident about the application of the proposed tool to real engineering problems.

Once arrived to this point, we can say that our embedded FSI framework arises as a possible alternative to solve the problems at hand. We would like to stress the achievements concerning thin-walled structures. It is known that these structures are prone to suffer from extremely large displacements that likely turn into topology changes when self-contact and wrinkling appear. In

view of the obtained results and according to the nature of the method, we think that our proposal has the potential of being an alternative to efficiently handle such challenging scenario.

As a summary, we collect in the following lines the main features of our proposal. The first one is that it does not need to reconstruct the system matrix graph each time the level set is updated. Besides that, it is purely local for both volume and volumeless bodies. Hence, any computational overhead coming from the neighbours calculation is completely avoided. Another feature that distinguishes our approach from other non-conforming alternatives is that we do the FSI interface BC imposition (both for volume and volumeless bodies) over the zero isosurface of the level set function, which is the best implicit representation of the real interface. Complementary, we would like to mention that, up to our knowledge, there is no other work in the literature that presents a strongly coupled embedded FSI framework conceived in an object oriented black-box fashion.

Nonetheless, we also note some disadvantages after comparing our technique to the standard body fitted approach. The main drawback is of course the lack of resolution in the vicinities of the immersed bodies. With this regard, we observe that ALE body fitted meshes normally yield a more accurate solution in the FSI interface than the one obtained with a similar embedded discretization. As this is a direct consequence of the nature of the method, which relies on solving the problem using an approximation of the discrete geometry, details can only be recovered upon mesh refinement. In short, we notice that for a similar resolution level, embedded methods normally require a finer mesh in the structure surroundings, which turns into a computational overhead.

We conclude the paper by briefly describing the further investigations arising from this work. The first task that is left to be done is the distributed memory implementation of the presented tool. Although both the embedded CFD and the CSM solvers are already compatible with MPI parallelism, we still lack the the FSI coupling implementation in order to efficiently solve real-life problems. Alternatively, we also plan to explore the possibilities of adaptive mesh refinement to tackle the reported mesh resolution issues. Last but not least, we also want to further work on the discontinuous level set representation of volumeless bodies. As we report in the 4-point tent example, it still has some issues in the embedded skin boundaries. We believe that these could be efficiently located and handled by using a discontinuous edge-based level set function instead of the current elemental one.

Acknowledgements

This research has been partly supported by the International Graduate School of Science and Engineering (IGSSE) and the European Comission (EC) through the projects ATMOPACE and ExaQUte (H2020-FETHPC-2016-2017-800898). The authors also acknowledge financial support from the Spanish Ministry of Economy and Competitiveness, through the "Severo Ochoa Programme for Centres of Excellence in R&D" (CEX2018-000797-S). Rubén Zor-

rilla gratefully acknowledges the support of the Spanish ministry for his FPU grant (FPU15/03796).

References

References

[1] R. Scotta, M. Lazzari, E. Stecca, J. Cotela, R. Rossi, Numerical wind tunnel for aerodynamic and aeroelastic characterization of bridge deck sections, Computers & Structures 167 (2016) 96 – 114 (2016). doi:10.1016/j.compstruc.2016.01.012.

[2] M. Saeedi, R. Wüchner, K.-U. Bletzinger, Fluid-structure interaction analysis and performance evaluation of a membrane blade, Journal of Physics: Conference Series 753 (2016) 102009 (sep 2016). doi:10.1088/1742-6596/753/10/102009.

[3] R. Mittal, H. Dong, M. Bozkurttas, F. Najjar, A. Vargas, A. von Loebbecke, A versatile sharp interface immersed boundary method for incompressible flows with complex boundaries, Journal of Computational Physics 227 (10) (2008) 4825 – 4852 (2008). doi:10.1016/j.jcp.2008.01.028.

[4] H. Zhao, J. B. Freund, R. D. Moser, A fixed-mesh method for incompressible flow–structure systems with finite solid deformations, Journal of Computational Physics 227 (6) (2008) 3114 – 3140 (2008). doi:10.1016/j.jcp.2007.11.019.

[5] C. A. Figueroa, I. E. Vignon-Clementel, K. E. Jansen, T. J. Hughes, C. A. Taylor, A coupled momentum method for modeling blood flow in three-dimensional deformable arteries, Computer Methods in Applied Mechanics and Engineering 195 (41) (2006) 5685 – 5706, john H. Argyris Memorial Issue. Part II (2006). doi:10.1016/j.cma.2005.11.011.

[6] P. Ryzhakov, E. Soudah, N. Dialami, Computational modeling of the fluid flow and the flexible intimal flap in type b aortic dissection via a monolithic arbitrary lagrangian/eulerian fluid-structure interaction model, International Journal for Numerical Methods in Biomedical Engineering n/a (n/a) (2019) e3239, e3239 cnm.3239 (2019). doi:10.1002/cnm.3239.

[7] L. Taelman, J. Bols, J. Degroote, V. Muthurangu, J. Panzer, J. Vierendeels, P. Segers, Differential impact of local stiffening and narrowing on hemodynamics in repaired aortic coarctation: an fsi study, Medical & Biological Engineering & Computing 54 (2) (2016) 497–510 (Mar 2016). doi:10.1007/s11517-015-1336-1.

[8] R. Zorrilla, E. Soudah, R. Rossi, Computational modeling of the fluid flow in type b aortic dissection using a modified finite element embedded formulation, Biomechanics and Modeling in Mechanobiology (Jan 2020). doi:10.1007/s10237-020-01291-x.

[9] S. Dahl, J. Vierendeels, J. Degroote, S. Annerel, L. Hellevik, B. Skallerud, Fsi simulation of asymmetric mitral valve dynamics during diastolic filling, Computer Methods in Biomechanics and Biomedical Engineering 15 (2) (2012) 121–130 (2012). doi:10.1080/10255842.2010.517200.

[10] D. Kamensky, M.-C. Hsu, D. Schillinger, J. Evans, A. Aggarwal, Y. Bazilevs, M. Sacks, T. Hughes, An immersogeometric variational framework for fluid–structure interaction: Application to bioprosthetic heart valves, Computer Methods in Applied Mechanics and Engineering 284 (2015) 1005 – 1053 (2015). doi:10.1016/j.cma.2014.10.040.

[11] M. de Tullio, G. Pascazio, A moving-least-squares immersed boundary method for simulating the fluid–structure interaction of elastic bodies with arbitrary thickness, Journal of Computational Physics 325 (2016) 201 – 225 (2016). doi:10.1016/j.jcp.2016.08.020.

[12] H. Casquero, Y. J. Zhang, C. Bona-Casas, L. Dalcin, H. Gomez, Non-body-fitted fluid–structure interaction: Divergence-conforming b-splines, fully-implicit dynamics, and variational formulation, Journal of Computational Physics 374 (2018) 625 – 653 (2018). doi:10.1016/j.jcp.2018.07.020.

[13] C. Hirt, A. Amsden, J. Cook, An arbitrary lagrangian-eulerian computing method for all flow speeds, Journal of Computational Physics 14 (3) (1974) 227–253 (1974). doi:10.1016/0021-9991(74)90051-5.

[14] J. Donea, A. Huerta, J.-P. Ponthot, A. Rodríguez-Ferran, Arbitrary Lagrangian–Eulerian Methods, American Cancer Society, 2004, Ch. 14 (2004). doi:10.1002/0470091355.ecm009.

[15] C. Peskin, Numerical analysis of blood flow in the heart, Journal of Computational Physics 25 (3) (1977) 220 – 252 (1977). doi:10.1016/0021-9991(77)90100-0.

[16] D. A. Field, Laplacian smoothing and delaunay triangulations, Communications in Applied Numerical Methods 4 (6) (1988) 709–712 (1988). doi:10.1002/cnm.1630040603.

[17] K. Stein, T. Tezduyar, R. Benney, Mesh Moving Techniques for Fluid-Structure Interactions With Large Displacements , Journal of Applied Mechanics 70 (1) (2003) 58–63 (01 2003). doi:10.1115/1.1530635.

[18] E. Oñate, S. R. Idelsohn, M. A. Celigueta, R. Rossi, Advances in the particle finite element method for the analysis of fluid–multibody interaction and bed erosion in free surface flows, Computer Methods in Applied Mechanics and Engineering 197 (19) (2008) 1777 – 1800, computational Methods in Fluid–Structure Interaction (2008). doi:10.1016/j.cma.2007.06.005.

[19] P. B. Ryzhakov, R. Rossi, S. R. Idelsohn, E. Oñate, A monolithic lagrangian approach for fluid–structure interaction problems, Computational Mechanics 46 (6) (2010) 883–899 (Nov 2010). doi:10.1007/s00466-010-0522-0.

[20] A. Franci, E. Oñate, J. M. Carbonell, Unified lagrangian formulation for solid and fluid mechanics and fsi problems, Computer Methods in Applied Mechanics and Engineering 298 (2016) 520 – 547 (2016). doi:10.1016/j.cma.2015.09.023.

[21] C. Peskin, The immersed boundary method, Acta Numerica 112 (2002) 479 – 517 (2002). doi:10.1017/S0962492902000077.

[22] E. Burman, S. Claus, P. Hansbo, M. G. Larson, A. Massing, Cutfem: Discretizing geometry and partial differential equations, International Journal for Numerical Methods in Engineering 104 (7) (2015) 472–501 (2015). doi:10.1002/nme.4823.

[23] A. Main, G. Scovazzi, The shifted boundary method for embedded domain computations. part i: Poisson and stokes problems, Journal of Computational Physics 372 (2018) 972 – 995 (2018). doi:10.1016/j.jcp.2017.10.026.

[24] A. Main, G. Scovazzi, The shifted boundary method for embedded domain computations. part ii: Linear advection–diffusion and incompressible navier–stokes equations, Journal of Computational Physics 372 (2018) 996 – 1026 (2018). doi:10.1016/j.jcp.2018.01.023.

[25] C. Peskin, Flow patterns around heart valves: A numerical method, Journal of Computational Physics 10 (2) (1972) 252 – 271 (1972). doi:10.1016/0021-9991(72)90065-4.

[26] C. Peskin, D. McQueen, A three-dimensional computational method for blood flow in the heart i. immersed elastic fibers in a viscous incompressible fluid, Journal of Computational Physics 81 (2) (1989) 372 – 405 (1989). doi:10.1016/0021-9991(89)90213-1.

[27] S.Osher, R. Fedkiw, Level Set Methods and Dynamic Implicit Surfaces, 1st Edition, Vol. 153 of Applied Mathematical Siences, Springer-Verlag New York, 2003 (2003).

[28] R. Zorrilla, A. Larese, R. Rossi, A modified finite element formulation for the imposition of the slip boundary condition over embedded volumeless geometries, Computer Methods in Applied Mechanics and Engineering 353 (2019) 123 – 157 (2019). doi:10.1016/j.cma.2019.05.007.

[29] D. Baumgärtner, J. Wolf, R. Rossi, P. Dadvand, R. Wüchner, A robust algorithm for implicit description of immersed geometries within a background mesh, Advanced Modeling and Simulation in Engineering Sciences 5 (1) (2018) 21 (Aug 2018). doi:10.1186/s40323-018-0113-8.

[30] J. Wolf, D. Baumgärtner, R. Rossi, P. Dadvand, R. Wüchner, Contribution to the Fluid-Structure Interaction Analysis of Ultra-Lightweight Structures using an Embedded Approach, International Center for Numerical Methods in Engineering, 2015 (01 2015). doi:10.13140/RG.2.1.1079.8561.

[31] J. Kim, D. Kim, H. Choi, An immersed-boundary finite-volume method for simulations of flow in complex geometries, Journal of Computational Physics 171 (1) (2001) 132 – 150 (2001). doi:https://doi.org/10.1006/jcph.2001.6778.

[32] L. Zhang, A. Gerstenberger, X. Wang, W. K. Liu, Immersed finite element method, Computer Methods in Applied Mechanics and Engineering 193 (21) (2004) 2051 – 2067, flow Simulation and Modeling (2004). doi:10.1016/j.cma.2003.12.044.

[33] Z.-Q. Zhang, G. R. Liu, B. C. Khoo, A three dimensional immersed smoothed finite element method (3d is-fem) for fluid–structure interaction problems, Computational Mechanics 51 (2) (2013) 129–150 (Feb 2013). doi:10.1007/s00466-012-0710-1.

[34] R. Löhner, J. D. Baum, E. Mestreau, D. Sharov, C. Charman, D. Pelessone, Adaptive embedded unstructured grid methods, International Journal for Numerical Methods in Engineering 60 (3) (2004) 641–660 (2004). doi:10.1002/nme.978.

[35] L. Foucard, F. Vernerey, An x-fem-based numerical–asymptotic expansion for simulating a stokes flow near a sharp corner, International Journal for Numerical Methods in Engineering 102 (2) (2015) 79–98 (2015). doi:10.1002/nme.4746.

[36] T. Sawada, A. Tezuka, Llm and x-fem based interface modeling of fluid–thin structure interactions on a non-interface-fitted mesh, Computational Mechanics 48 (3) (2011) 319–332 (Sep 2011). doi:10.1007/s00466-011-0600-y.

[37] F. Alauzet, B. Fabrèges, M. A. Fernández, M. Landajuela, Nitsche-xfem for the coupling of an incompressible fluid with immersed thin-walled structures, Computer Methods in Applied Mechanics and Engineering 301 (2016) 300 – 335 (2016). doi:10.1016/j.cma.2015.12.015.

[38] J. Nitsche, Über ein variationsprinzip zur lösung von dirichlet-problemen bei verwendung von teilräumen, die keinen randbedingungen unterworfen sind, Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg 36 (1) (1971) 9–15 (Jul 1971). doi:10.1007/BF02995904.

[39] J. Baiges, R. Codina, F. Henke, S. Shahmiri, W. A. Wall, A symmetric method for weakly imposing dirichlet boundary conditions in embedded finite element meshes, International Journal for Numerical Methods in Engineering 90 (5) (2012) 636–658 (2012). doi:10.1002/nme.3339.

[40] R. Codina, J. Baiges, Approximate imposition of boundary conditions in immersed boundary methods, Int. J. Numer. Meth. Engng. 80 (2009) 1379–1405 (2009). doi:10.1002/nme.2662.

[41] A. Massing, M. Larson, A. Logg, M. Rognes, A stabilized nitsche fictitious domain method for the stokes problem, Journal of Scientific Computing 61 (3) (2014) 604–628 (Dec 2014). doi:10.1007/s10915-014-9838-9.

[42] A. Massing, B. Schott, W. Wall, A stabilized nitsche cut finite element method for the oseen problem, Comput. Methods Appl. Mech. Engrg. 328 (2018) 262–300 (2018). doi:10.1016/j.cma.2017.09.003.

[43] J. Urquiza, A. Garon, M.-I. Farinas, Weak imposition of the slip boundary condition on curved boundaries for stokes flow, Journal of Computational Physics 256 (2014) 748–767 (2014). doi:10.1016/j.jcp.2013.08.045.

[44] M. Winter, B. Schott, A. Massing, W. Wall, A nitsche cut finite element method for the oseen problem with general navier boundary conditions, Comput. Methods Appl. Mech. Engrg. 330 (2018) 220–252 (2018). doi:10.1016/j.cma.2017.10.023.

[45] R. Ausas, F. Sousa, G. Buscaglia, An improved finite element space for discontinuous pressures, Comput. Methods Appl. Mech. Engrg. 199 (2010) 1019–1031 (2010). doi:10.1016/j.cma.2009.11.011.

[46] R. Zorrilla, A. Larese, R. Rossi, A discontinuous nitsche based finite element formulation for the imposition of the general navier-slip boundary condition over embedded volumeless geometries, Computer Methods in Applied Mechanics and Engineering (under review).

[47] S. Badia, F. Verdugo, A. F. Martín, The aggregated unfitted finite element method for elliptic problems, Computer Methods in Applied Mechanics and Engineering 336 (2018) 533 – 553 (2018). doi:10.1016/j.cma.2018.03.022.

[48] R. Codina, G. Houzeaux, H. Coppola-Owen, J. Baiges, The fixed-mesh ale approach for the numerical approximation of flows in moving domains, Journal of Computational Physics 228 (5) (2009) 1591 – 1611 (2009). doi:10.1016/j.jcp.2008.11.004.

[49] J. Baiges, R. Codina, The fixed-mesh ale approach applied to solid mechanics and fluid–structure interaction problems, International Journal for Numerical Methods in Engineering 81 (12) (2010) 1529–1557 (2010). doi:10.1002/nme.2740.

[50] J. Baiges, R. Codina, H. Coppola-Owen, The fixed-mesh ale approach for the numerical simulation of floating solids, International Journal for Numerical Methods in Fluids 67 (8) (2011) 1004–1023 (2011). doi:10.1002/fld.2403.

[51] R. Rossi, E. Oñate, Analysis of some partitioned algorithms for fluid-structure interaction, Engineering Computations 27 (1) (2010) 20–56 (Jan 2010). doi:10.1108/02644401011008513.

[52] S. Badia, F. Nobile, C. Vergara, Fluid–structure partitioned procedures based on robin transmission conditions, Journal of Computational Physics 227 (14) (2008) 7027 – 7051 (2008). doi:10.1016/j.jcp.2008.04.006.

[53] L. Gerardo-Giorda, F. Nobile, C. Vergara, Analysis and optimization of robin–robin partitioned procedures in fluid–structure interaction problems, SIAM Journal on Numerical Analysis 48 (6) (2010) 2091–2116 (2010). doi:10.1137/09076605X.

[54] M. A. Fernández, M. Landajuela, M. Vidrascu, Fully decoupled time-marching schemes for incompressible fluid/thin-walled structure interaction, Journal of Computational Physics 297 (2015) 156 – 181 (2015). doi:10.1016/j.jcp.2015.05.009.

[55] M. Landajuela, M. Vidrascu, D. Chapelle, M. A. Fernández, Coupling schemes for the fsi forward prediction challenge: Comparative study and validation, International Journal for Numerical Methods in Biomedical Engineering 33 (4) (2017) e2813, e2813 cnm.2813 (2017). doi:10.1002/cnm.2813.

[56] R. Rossi, P. B. Ryzhakov, E. Oñate, A monolithic fe formulation for the analysis of membranes in fluids, International Journal of Space Structures 24 (4) (2009) 205–210 (2009). doi:10.1260/026635109789968263.

[57] C. Farhat, K. G. van der Zee, P. Geuzaine, Provably second-order time-accurate loosely-coupled solution algorithms for transient nonlinear computational aeroelasticity, Computer Methods in Applied Mechanics and Engineering 195 (17) (2006) 1973 – 2001, fluid-Structure Interaction (2006). doi:10.1016/j.cma.2004.11.031.

[58] P. Causin, J. Gerbeau, F. Nobile, Added-mass effect in the design of partitioned algorithms for fluid–structure problems, Computer Methods in Applied Mechanics and Engineering 194 (42) (2005) 4506 – 4527 (2005). doi:10.1016/j.cma.2004.12.005.

[59] S. R. Idelsohn, F. Del Pin, R. Rossi, E. Oñate, Fluid–structure interaction problems with strong added-mass effect, International Journal for Numerical Methods in Engineering 80 (10) (2009) 1261–1294 (2009). doi:10.1002/nme.2659.

[60] J. Degroote, K.-J. Bathe, J. Vierendeels, Performance of a new partitioned procedure versus a monolithic procedure in fluid–structure interaction, Computers & Structures 87 (11) (2009) 793 – 801, fifth MIT Conference on Computational Fluid and Solid Mechanics (2009). doi:https://doi.org/10.1016/j.compstruc.2008.11.013.

[61] J. Degroote, J. Vierendeels, Multi-solver algorithms for the partitioned simulation of fluid–structure interaction, Computer Methods in Applied Mechanics and Engineering 200 (25) (2011) 2195 – 2210 (2011). doi:10.1016/j.cma.2011.03.015.

[62] A. Bogaers, S. Kok, B. Reddy, T. Franz, Quasi-newton methods for implicit black-box fsi coupling, Computer Methods in Applied Mechanics and Engineering 279 (2014) 113 – 132 (2014). doi:10.1016/j.cma.2014.06.033.

[63] S. Sicklinger, V. Belsky, B. Engelmann, H. Elmqvist, H. Olsson, R. Wüchner, K.-U. Bletzinger, Interface jacobian-based co-simulation, International Journal for Numerical Methods in Engineering 98 (6) (2014) 418–444 (2014). doi:10.1002/nme.4637.

[64] U. Küttler, W. A. Wall, Fixed-point fluid–structure interaction solvers with dynamic relaxation, Computational Mechanics 43 (1) (2008) 61–72 (Dec 2008). doi:10.1007/s00466-008-0255-5.

[65] D. Mok, W. Wall, E. Ramm, Accelerated iterative substructuring schemes for instationary fluid-structure interaction, Computational fluid and solid mechanics 2 (2001) 1325–1328 (2001).

[66] D. Knoll, D. Keyes, Jacobian-free newton–krylov methods: a survey of approaches and applications, Journal of Computational Physics 193 (2) (2004) 357 – 397 (2004). doi:10.1016/j.jcp.2003.08.010.

[67] U. Küttler, M. Gee, C. Förster, A. Comerford, W. A. Wall, Coupling strategies for biomedical fluid–structure interaction problems, International Journal for Numerical Methods in Biomedical Engineering 26 (3–4) (2010) 305–321 (2010). doi:10.1002/cnm.1281.

[68] S. Minami, S. Yoshimura, Performance evaluation of nonlinear algorithms with line-search for partitioned coupling techniques for fluid–structure interactions, International Journal for Numerical Methods in Fluids 64 (10–12) (2010) 1129–1147 (2010). doi:10.1002/fld.2274.

[69] J. Degroote, R. Haelterman, S. Annerel, P. Bruggeman, J. Vierendeels, Performance of partitioned procedures in fluid–structure interaction, Computers & Structures 88 (7) (2010) 446 – 457 (2010). doi:10.1016/j.compstruc.2009.12.006.

[70] T. Spenke, N. Hosters, M. Behr, A multi-vector interface quasi-newton method with linear complexity for partitioned fluid–structure interaction, Computer Methods in Applied Mechanics and Engineering 361 (2020) 112810 (2020). doi:https://doi.org/10.1016/j.cma.2019.112810.

[71] P. Dadvand, R. Rossi, E. Oñate, An object-oriented environment for developing finite element codes for multi-disciplinary applications, Archives of Computational Methods in Engineering 17 (3) (2010) 253–297 (2010). doi:10.1007/s11831-010-9045-2.

[72] P. Dadvand, R. Rossi, M. Gil, X. Martorell, J. Cotela, E. Juanpere, S. Idelsohn, E. Oñate, Migration of a generic multi-physics framework to HPC environments, Computers & Fluids 80 (2013) 301 – 309 (2013). doi:10.1016/j.compfluid.2012.02.004.

[73] A. Coll, R. Ribó, M. Pasenau, E. Escolano, J. Perez, A. Melendo, A. Monros, J. Gárate, GiD v.13 User Manual (2016).

[74] A. Coll, R. Ribó, M. Pasenau, E. Escolano, J. Perez, A. Melendo, A. Monros, J. Gárate, GiD v.13 Reference Manual (2016).

[75] T. Hughes, L. Franca, M. Balestra, A new finite element formulation for computational fluid dynamics: V. circumventing the babuška-brezzi condition: a stable petrov-galerkin formulation of the stokes problem accommodating equal-order interpolations, Computer Methods in Applied Mechanics and Engineering 59 (1) (1986) 85 – 99 (1986). doi:10.1016/0045-7825(86)90025-3.

[76] J. Cotela, R. Rossi, E. Oñate, A FIC-based stabilized finite element formulation for turbulent flows, Comput. Methods Appl. Mech. Engrg. 315 (2017) 607 – 631 (2017). doi:10.1016/j.cma.2016.11.020.

[77] T. Tezduyar, S. Mittal, S. Ray, R. Shih, Incompressible flow computations with stabilized bilinear and linear equal-order-interpolation velocity-pressure elements, Comput. Methods Appl. Mech. Engrg. 95 (2) (1992) 221–242 (1992). doi:10.1016/0045-7825(92)90141-6.

[78] T. Hughes, Multiscale phenomena: Green's function, the dirichlet to neumann formulation, subgrid scale models, bubbles and the origins of stabilized formulations, Comput. Methods Appl. Mech. Engrg. 127 (1) (1995) 387–401 (1995). doi:10.1016/0045-7825(95)00844-9.

[79] T. Hughes, G. Feijóo, L. Mazzei, J. Quincy, The variational multiscale method—a paradigm for computational mechanics, Comput. Methods Appl. Mech. Engrg. 166 (1) (1998) 3–24, advances in Stabilized Methods in Computational Mechanics (1998). doi:10.1016/S0045-7825(98)00079-6.

[80] R. Codina, A stabilized finite element method for generalized stationary incompressible flows, Comput. Methods Appl. Mech. Engrg. 190 (20) (2001) 2681–2706 (2001). doi:10.1016/S0045-7825(00)00260-7.

[81] T. Belytschko, W. K. Liu, B. Moran, K. Elkhodary, Nonlinear finite elements for continua and structures., John wiley & sons, 2013 (2013).

[82] P. Khosravi, R. Ganesan, R. Sedaghati, Corotational non-linear analysis of thin plates and shells using a new shell element, International Journal for Numerical Methods in Engineering 69 (4) (2007) 859–885 (2007). doi:10.1002/nme.1791.

[83] G. Valdés, Nonlinear analysis of orthotropic membrane and shell structures including fluid-structure interaction, Ph.D. thesis, Universitat Politècnica de Catalunya (10 2007).

[84] S. Turek, J. Hron, Proposal for numerical benchmarking of fluid-structure interaction between an elastic object and laminar incompressible flow, in: H. Bungartz, M. Schäfer (Eds.), Fluid-Structure Interaction, Springer Berlin Heidelberg, Berlin, Heidelberg, 2006, pp. 371–385 (2006). doi:10.1007/3-540-34596-5_15.

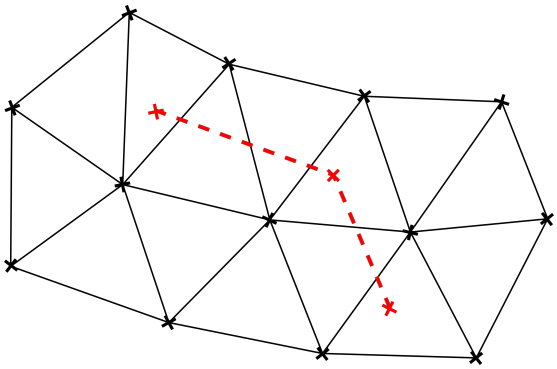
[85] C. A. Felippa, A study of optimal membrane triangles with drilling freedoms, Computer Methods in Applied Mechanics and Engineering 192 (16) (2003) 2125 – 2168 (2003). doi:10.1016/S0045-7825(03)00253-6.

[86] F. Otto, R. Trostel, F. K. Schleyer, Tensile structures: design, structure, and calculation of buildings of cables, nets, and membranes, Vol. 69, MIT Press London, 1967 (1967).

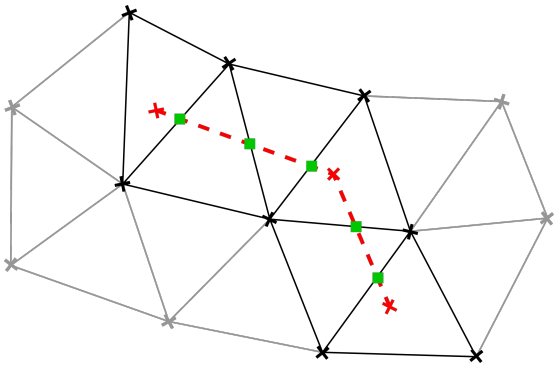
[87] E. Oñate, F. G. Flores, J. Marcipar, Membrane Structures Formed by Low Pressure Inflatable Tubes. New Analysis Methods and Recent Constructions, Springer Netherlands, Dordrecht, 2008, pp. 163–196 (2008). doi:10.1007/978-1-4020-6856-0_10.

[88] J. Hoffman, J. Jansson, C. Johnson, New theory of flight, Journal of Mathematical Fluid Mechanics 18 (10 2015). doi:10.1007/s00021-015-0220-y.

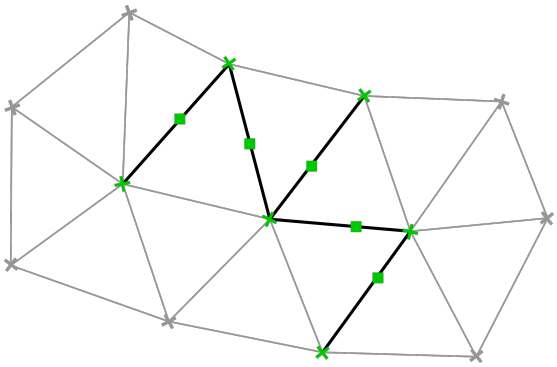
[89] J. Kiendl, Isogeometric analysis and shape optimal design of shell structures, Ph.D. thesis, Technische Universität München (2011).



(a) Toy example representation. Black solid lines represent the background mesh. Red dashed lines represent the embedded structure skin mesh.



(b) Intersected elements and edges search (black solid lines). Edge intersection points highlighted with light green square dots.



(c) Edge elements mesh to solve the extrapolation problem (black solid lines). The nodes of interest where the extrapolation is solved are highlighted in light green.

Figure 7: Embedded nodal variable extrapolation from skin to background mesh.

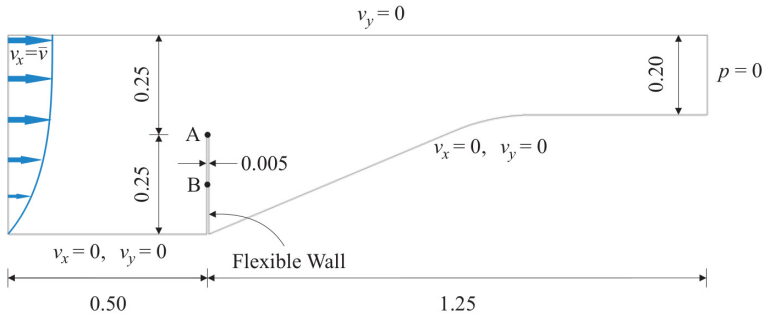
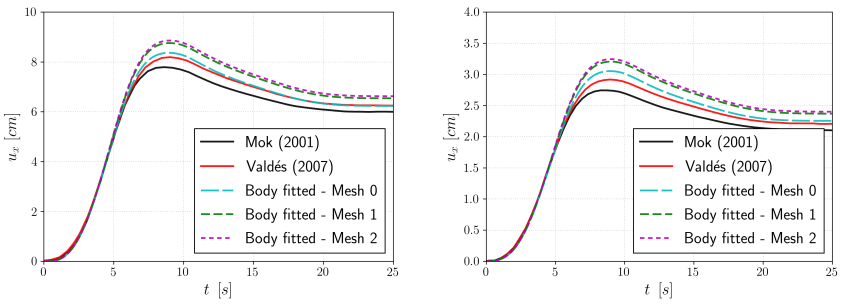


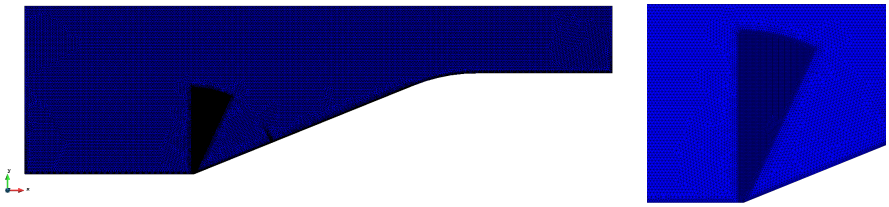
Figure 8: 2D Mok benchmark. Geometry and boundary conditions (source [83]).



(a) Point A.

(b) Point B.

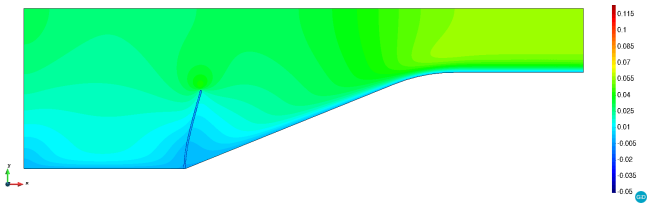
Figure 9: 2D Mok benchmark. Control points u_x [cm] convergence study.



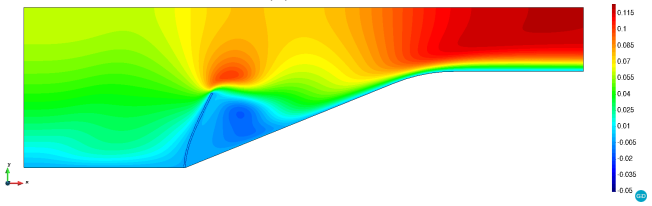
(a) Complete view.

(b) Zoom view.

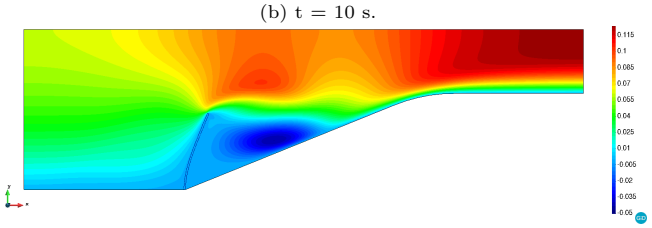
Figure 10: 2D Mok benchmark.Fluid background mesh snapshots.



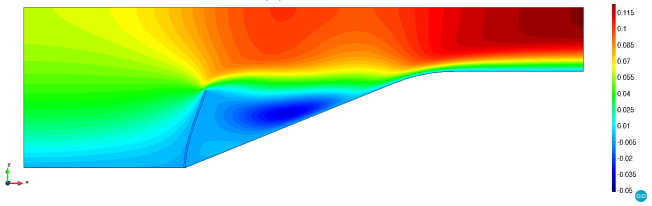
(a) t = 5 s.



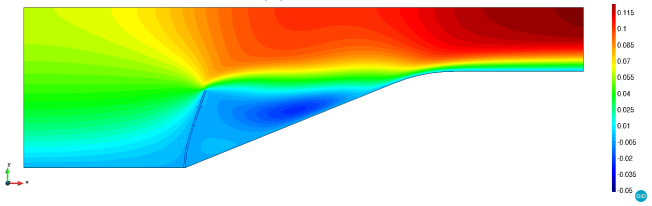
(b) t = 10 s.



(c) t = 15 s.

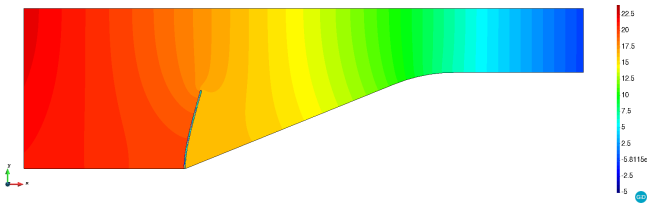


(d) t = 20 s.

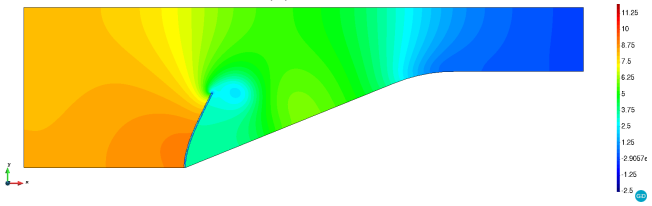


(e) t = 25 s.

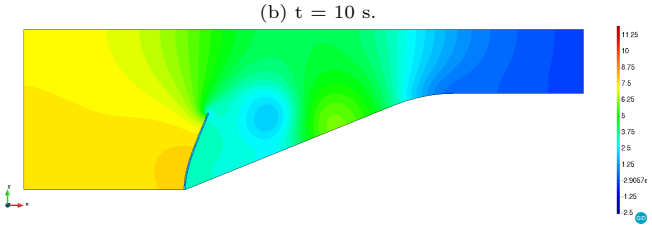
Figure 11: 2D Mok benchmark. Level set zero isosurface and v_x velocity field [m/s] at different time steps.



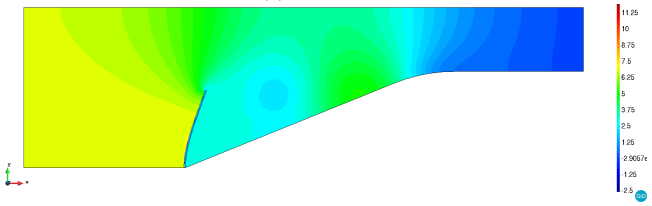
(a) t = 5 s.



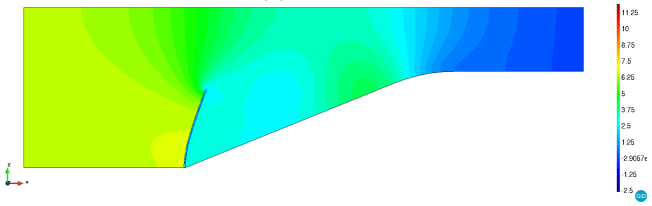
(b) t = 10 s.



(c) t = 15 s.

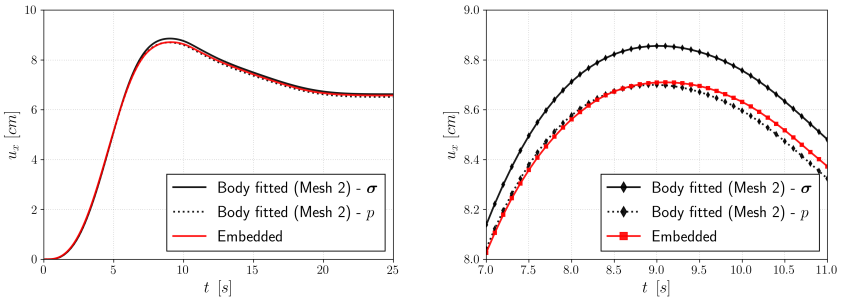


(d) t = 20 s.



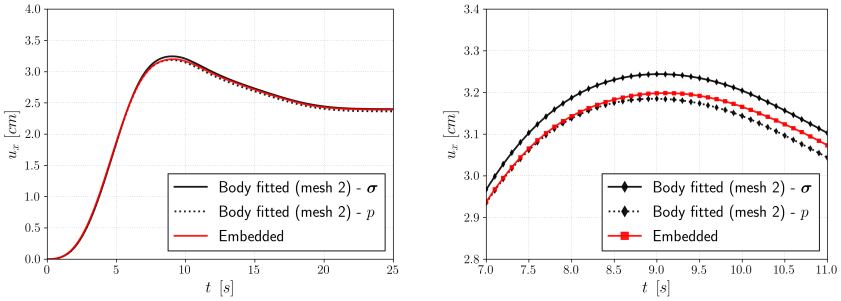
(e) t = 25 s.

Figure 12: 2D Mok benchmark. Level set zero isosurface and pressure field [Pa] at different time steps.



(a) Complete time history. (b) Zoom from 7 to 11s.

Figure 13: 2D Mok benchmark. Control point A horizontal displacement u_x [cm].



(a) Complete time history. (b) Zoom from 7 to 11s.

Figure 14: 2D Mok benchmark. Control point B horizontal displacement u_x [cm].

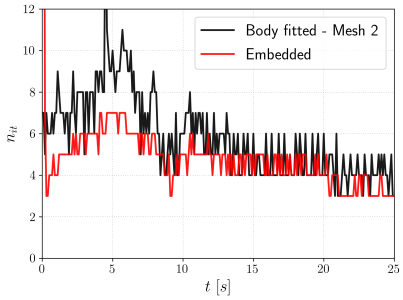
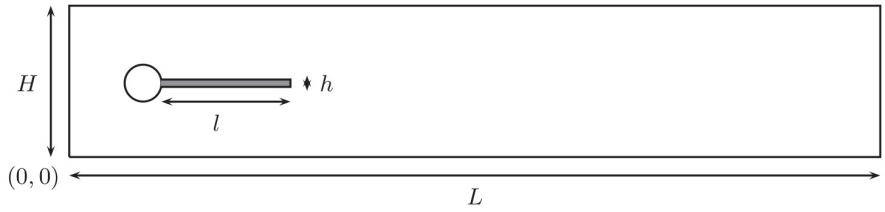
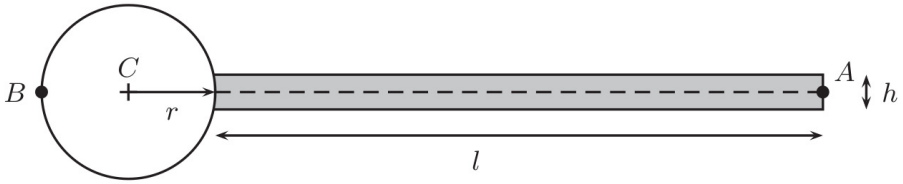


Figure 15: 2D Mok benchmark. FSI non-linear iterations history.

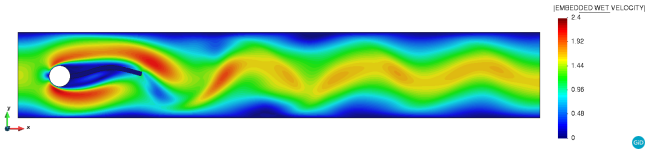


(a) Fluid domain.

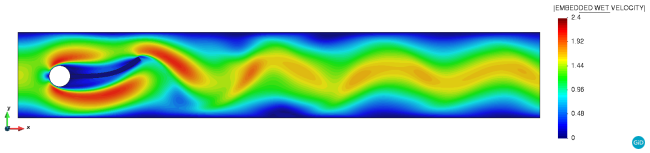


(b) Structure domain.

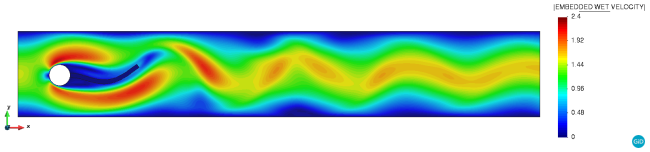
Figure 16: 2D Turek & Hron benchmark - FSI2. Problem geometry (source [84]).



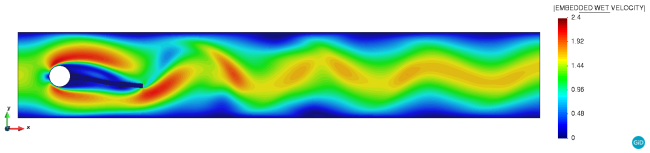
(a) t = t_0 s.



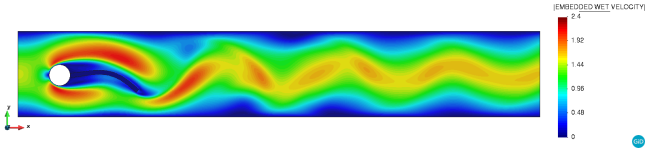
(b) t = t_0 + 0.2T s.



(c) t = t_0 + 0.4T s.

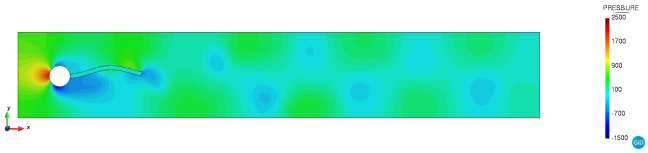


(d) t = t_0 + 0.6T s.

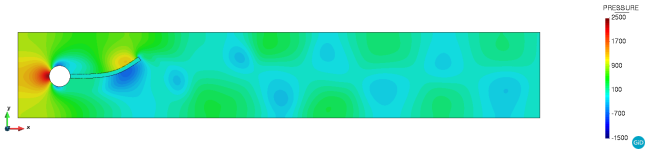


(e) t = t_0 + 0.8T s.

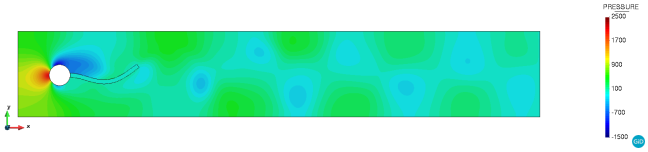
Figure 17: 2D Turek & Hron benchmark - FSI2. Level set zero isosurface and velocity norm $\|\mathbf{v}\|$ [m/s] for a period of oscillation T.



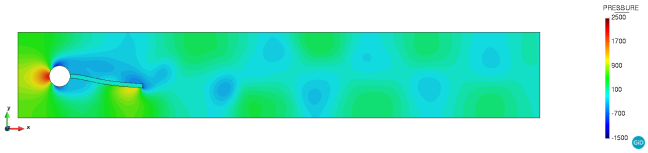
(a) t = t_0 s.



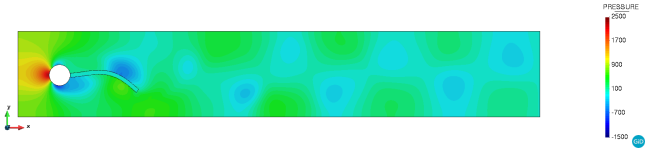
(b) t = t_0 + 0.2T s.



(c) t = t_0 + 0.4T s.

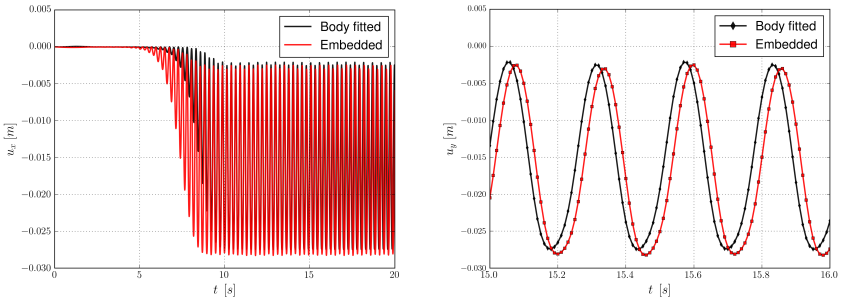


(d) t = t_0 + 0.6T s.



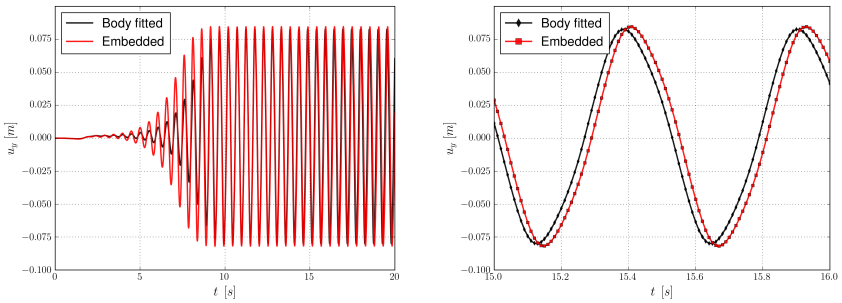
(e) t = t_0 + 0.8T s.

Figure 18: 2D Turek & Hron benchmark - FSI2. Level set zero isosurface and pressure p [Pa] for a period of oscillation T.



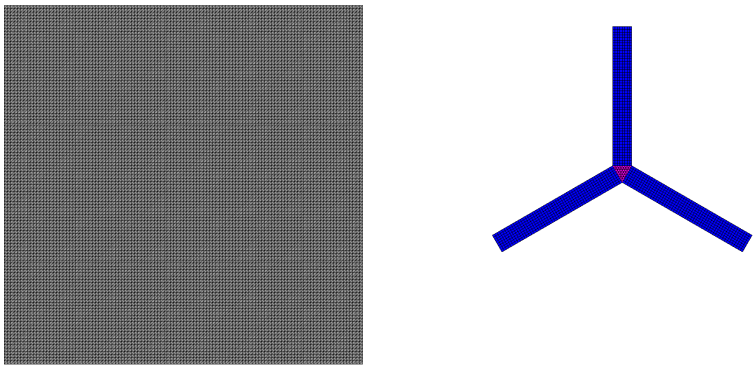
(a) Complete time history. (b) Zoom from 15 to 16s.

Figure 19: 2D Turek & Hron benchmark - FSI2. Control point A horizontal displacement u_x [m].



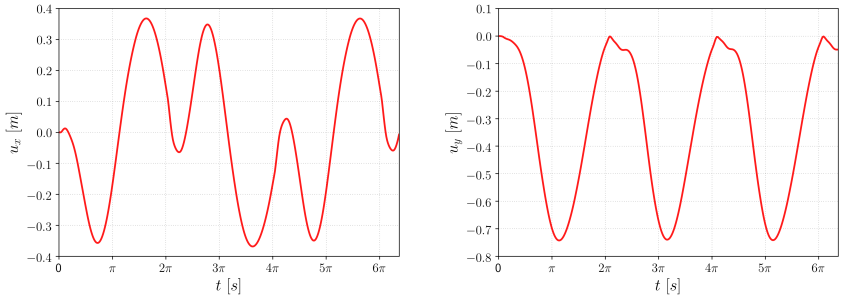
(a) Complete time history. (b) Zoom from 15 to 16s.

Figure 20: 2D Turek & Hron benchmark - FSI2. Control point A vertical displacement u_y [m].



(a) Fluid domain. (b) Structure domain.

Figure 21: 2D mixer with flexible blades. Problem geometry with mesh.



(a) u_x [m]. (b) u_y [m].

Figure 22: 2D mixer with flexible blades. Control point (0,0.375)m displacement evolution.

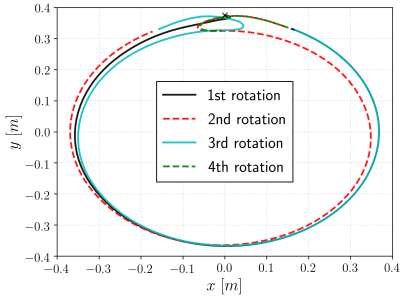
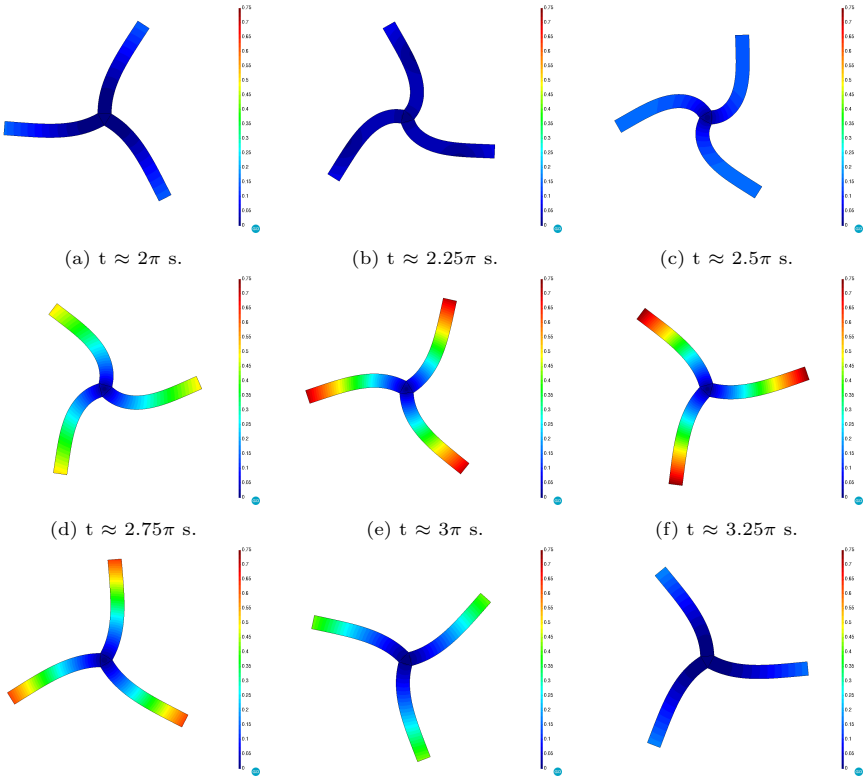


Figure 23: 2D mixer with flexible blades. Control point tracking. Solid lines represent the anticlockwise rotation cycles. Dashed lines represent the clockwise rotation ones. The black X marker represents the initial position of the control point.

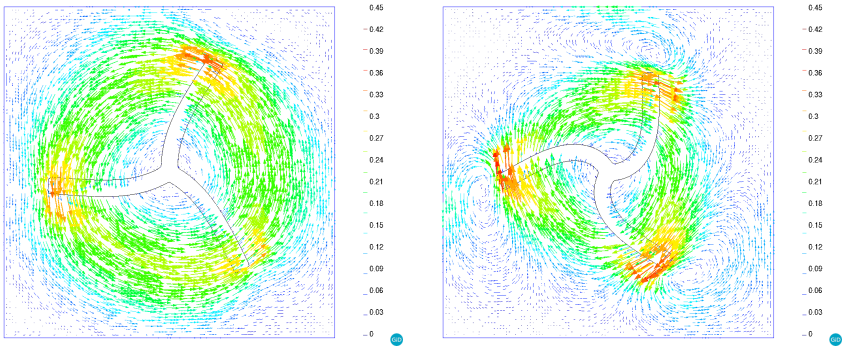


(a) t ≈ 2π s. (b) t ≈ 2.25π s. (c) t ≈ 2.5π s.

(d) t ≈ 2.75π s. (e) t ≈ 3π s. (f) t ≈ 3.25π s.

(g) t ≈ 3.5π s. (h) t ≈ 3.75π s. (i) t ≈ 4π s.

Figure 24: 2D mixer with flexible blades. Structure displacement norm $\|\mathbf{u}\|$ [m] snapshots during the second (clockwise) rotation ($2\pi \leq t \leq 4\pi$ [s]).



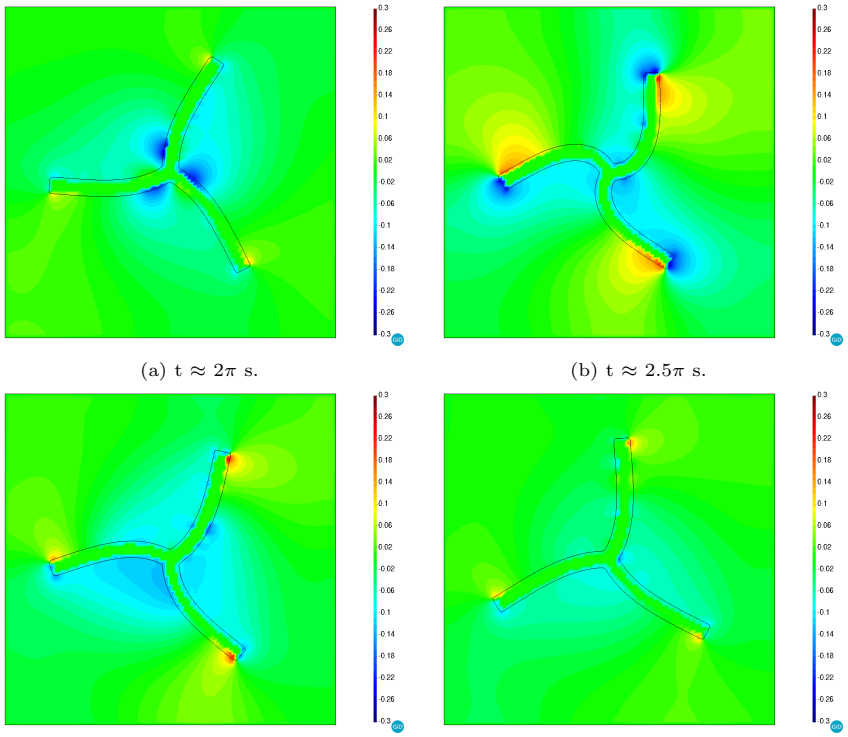
(a) t $\approx 2\pi$ s.

(b) t $\approx 2.5\pi$ s.

(c) t $\approx 3\pi$ s.

(d) t $\approx 3.5\pi$ s.

Figure 25: 2D mixer with flexible blades. Level set zero isosurface and velocity vector field [m/s] snapshots during the second (clockwise) rotation ($2\pi \leq t \leq 4\pi$ [s]).



(a) t $\approx 2\pi$ s.

(b) t $\approx 2.5\pi$ s.

(c) t $\approx 3\pi$ s.

(d) t $\approx 3.5\pi$ s.

Figure 26: 2D mixer with flexible blades. Level set zero isosurface and pressure field [Pa] snapshots during the second (clockwise) rotation ($2\pi \leq t \leq 4\pi$ $[s]$).

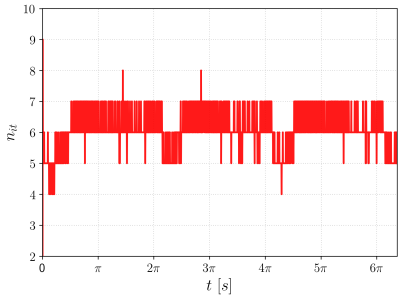
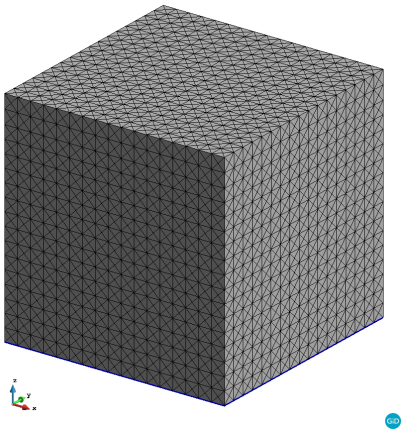
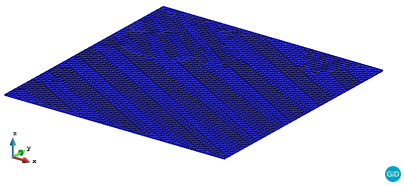


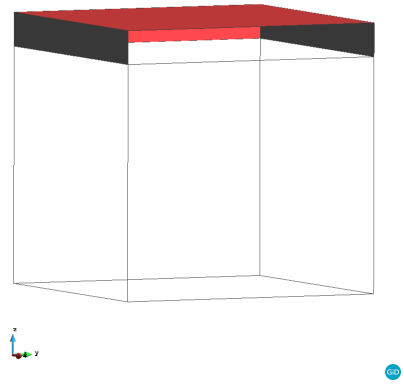
Figure 27: 2D mixer with flexible blades. FSI non-linear iterations history.



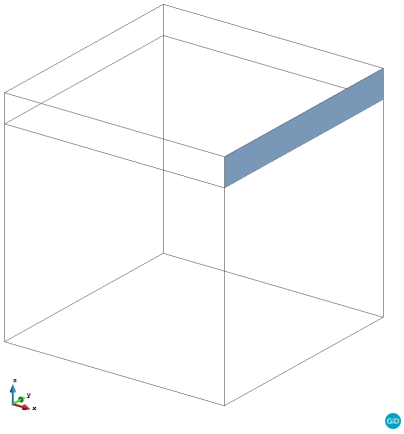
(a) Fluid cavity.



(b) Flexible shell.



(c) Imposed velocity surfaces.



(d) Fixed pressure surface.

Figure 28: 3D lid-driven cavity with flexible shell. Problem geometry.

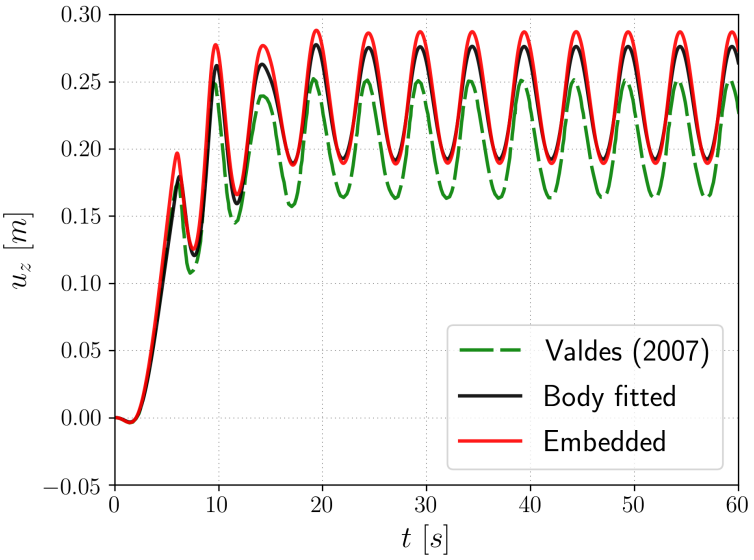
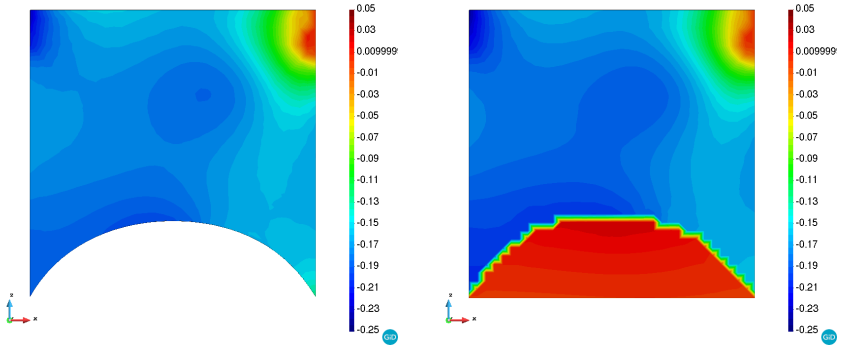


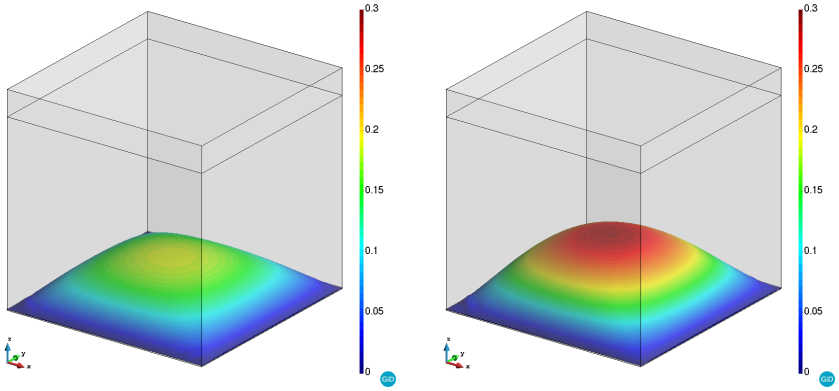
Figure 29: 3D lid-driven cavity with flexible shell. Midpoint (0.5,0.5)m vertical displacement u_z evolution [m].



(a) Body fitted.

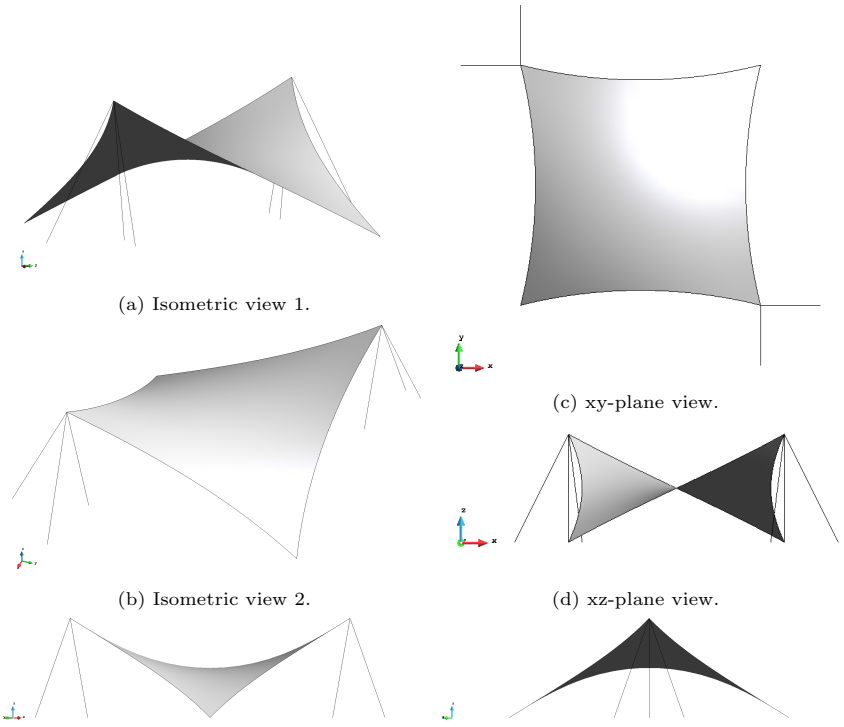
(b) Embedded.

Figure 30: 3D lid-driven cavity with flexible shell. Pressure field (Pa) xz-plane cross section at maximum vertical displacement u_z.



(a) t = 42.0s (min. u_z). (b) t = 44.4s (max. u_z).

Figure 31: 3D lid-driven cavity with flexible shell. Displacement norm $\|\mathbf{u}\|$ snapshots during periodic regime [m].



(a) Isometric view 1.

(c) xy-plane view.

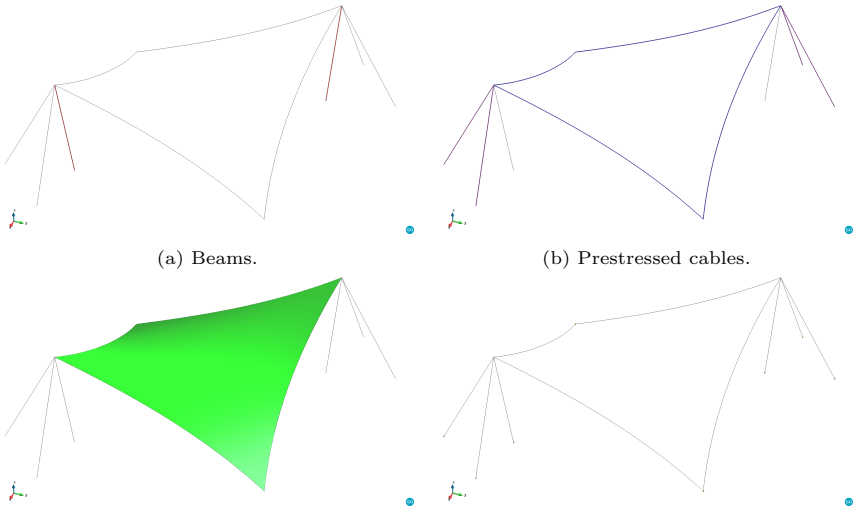
(b) Isometric view 2.

(d) xz-plane view.

(e) Lateral view 1.

(f) Lateral view 2.

Figure 32: 3D four-point tent. Structure geometry.



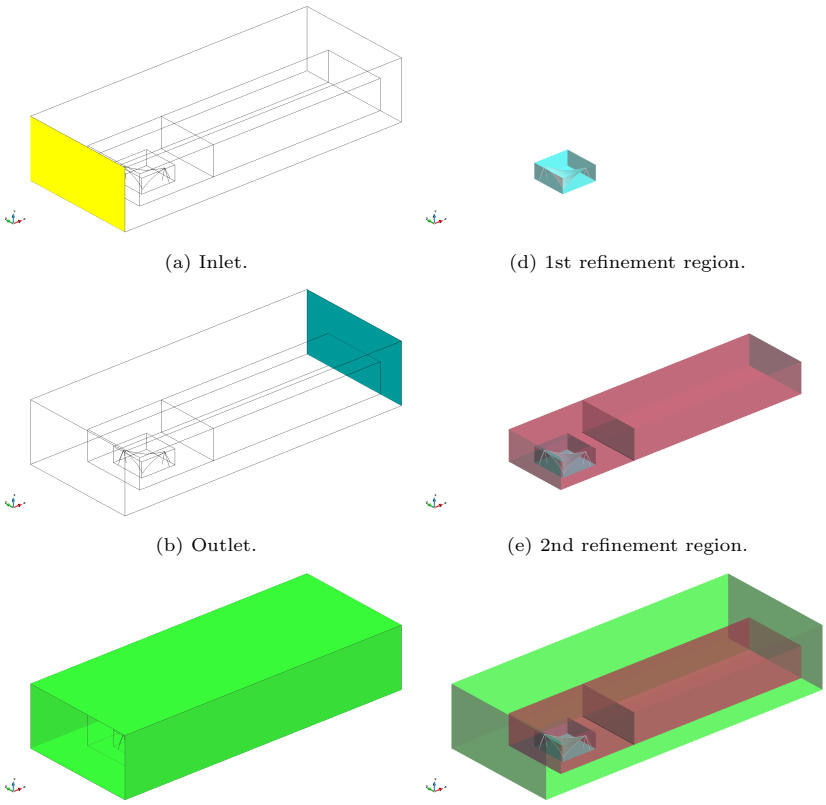
(a) Beams.

(b) Prestressed cables.

(c) Prestressed membrane.

(d) Dirichlet boundary conditions.

Figure 33: 3D four-point tent. Element types and boundary conditions. The beams are highlighted in red. The supporting cables are highlighted in purple while the membrane edges ones are highlighted in blue. The prestressed membrane is highlighted in light green. The light brown square dots represent the fixed displacement points.



(a) Inlet.

(d) 1st refinement region.

(b) Outlet.

(e) 2nd refinement region.

(c) Slip walls.

(f) 3rd refinement region.

Figure 34: 3D four-point tent. Fluid geometry, boundary conditions and refinement regions. The structure geometry is superimposed to the background mesh geometry for the sake of visualization.

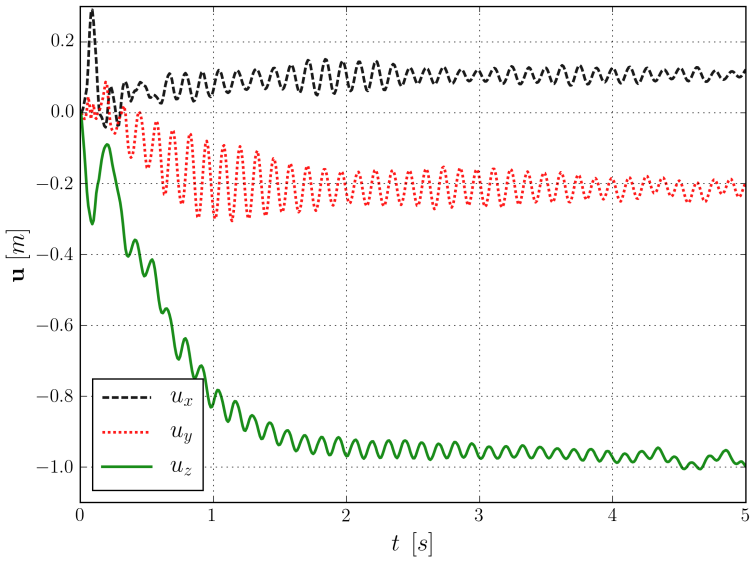
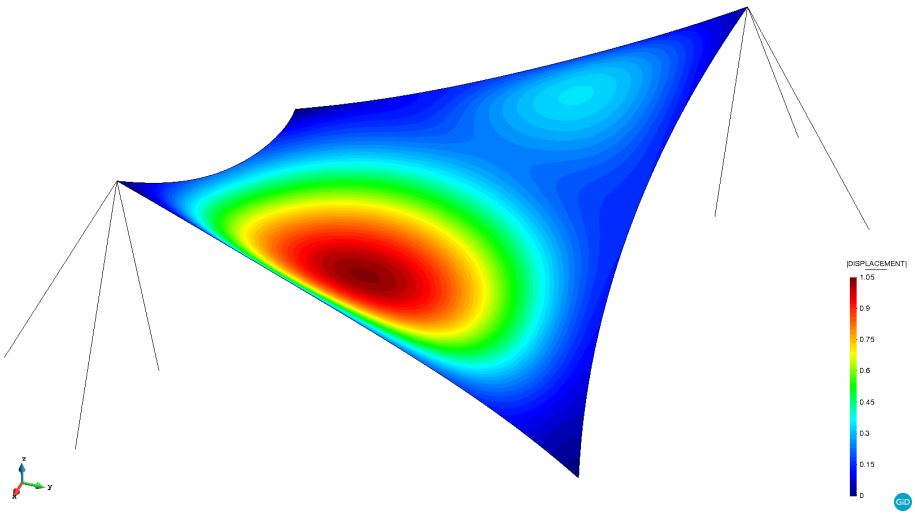
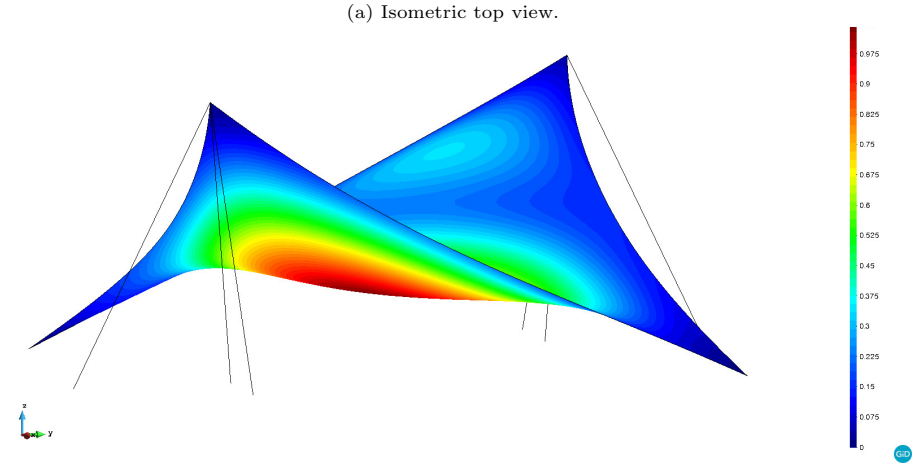


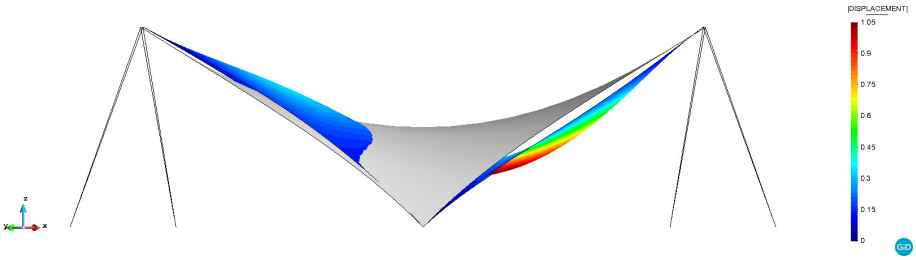
Figure 35: 3D four-point tent. Point (5.25,3.25) vertical displacement u_z evolution [m].



(a) Isometric top view.

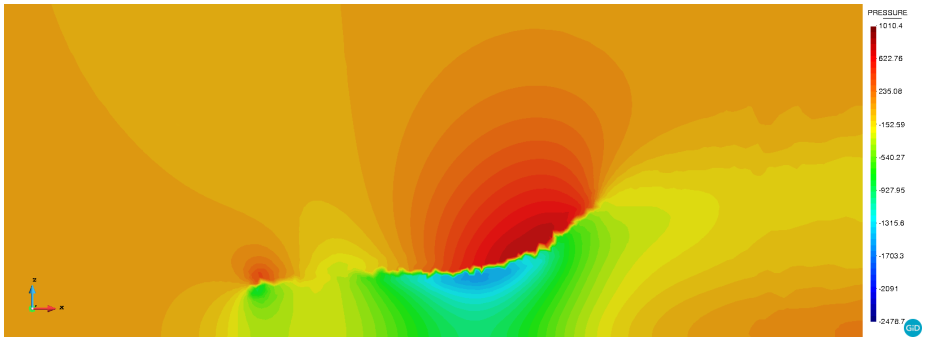


(b) Isometric rear view.

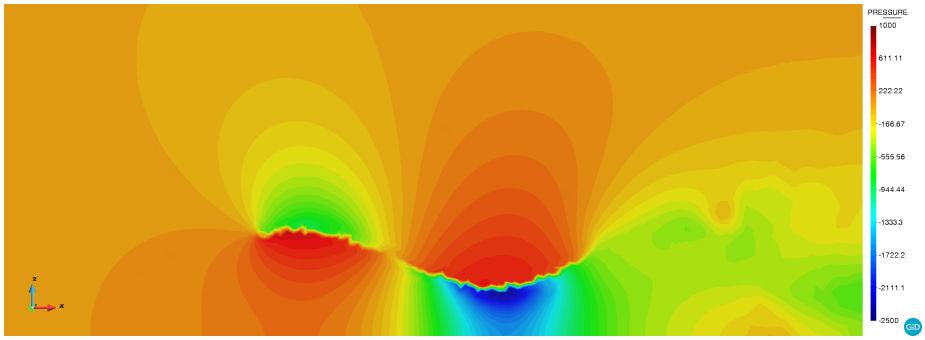


(c) Lateral view with undeformed configuration.

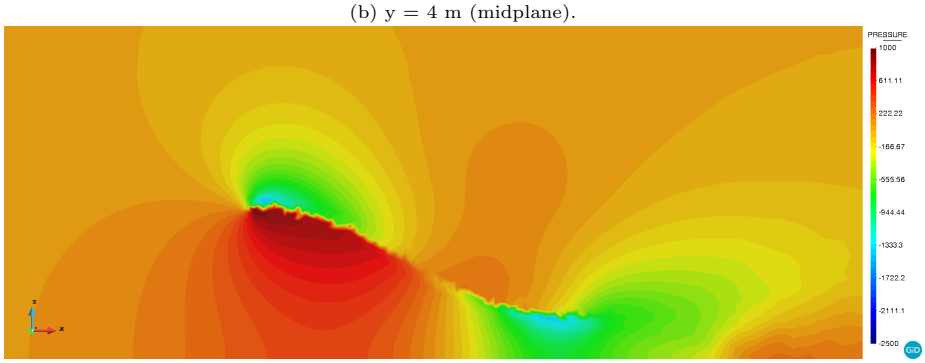
Figure 36: 3D four-point tent. Structure $\|\mathbf{u}\|$ displacement field at time t = 5 s.



(a) y = 2 m.

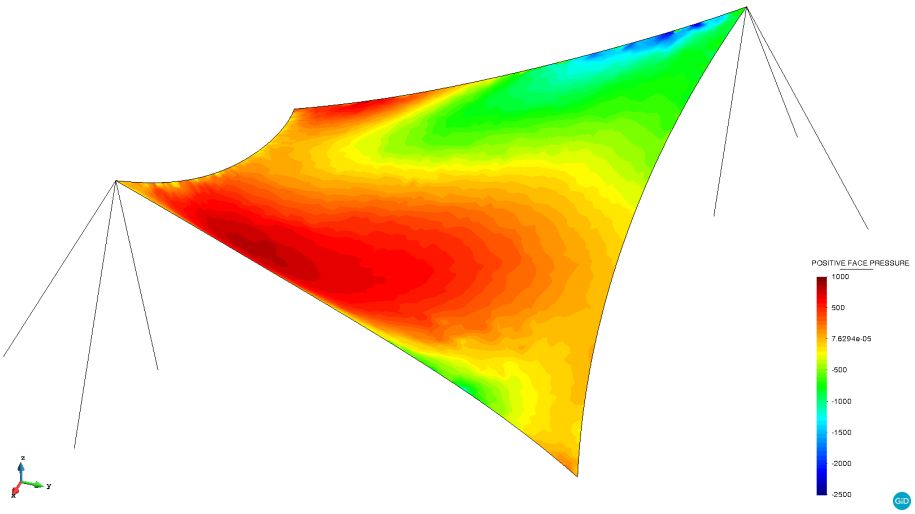


(b) y = 4 m (midplane).

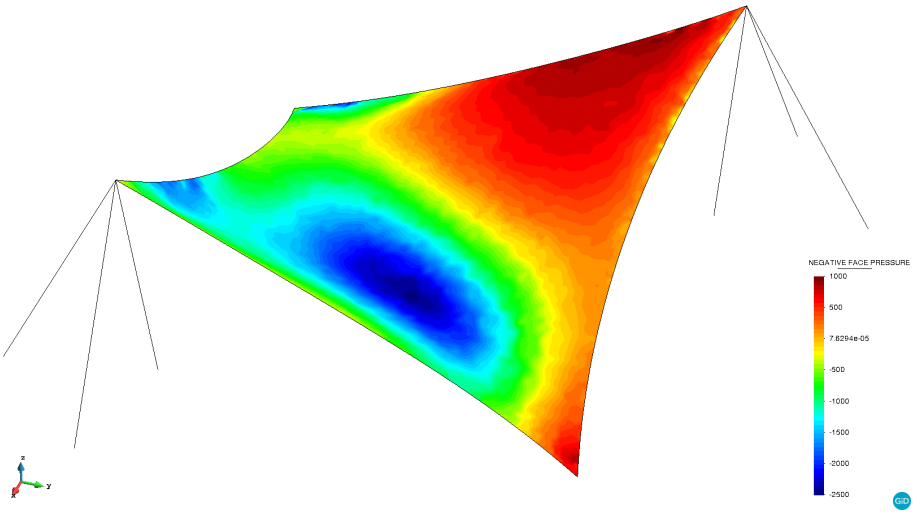


(c) y = 6 m.

Figure 37: 3D four-point tent. Pressure field xz-plane cross sections at time t = 5 s.



(a) Positive (top) membrane side.



(b) Negative (bottom) membrane side.

Figure 38: 3D four-point tent. Pressure over the prestressed membrane at time t = 5 s.