3D Nitsche-XFEM method for fluid-structure interaction with immersed thin-walled solids

Fannie Maria Gerosa, Daniele Corti, Frédéric Alauzet, Miguel Angel Fernández

3D Nitsche-XFEM method for fluid-structure interaction with immersed thin-walled solids

Fannie M. Gerosa[a,b], Daniele C. Corti[b], Frédéric Alauzet[c], Miguel A. Fernández[b,*]

[a] *Stanford University, Stanford, USA*
[b] *Sorbonne Université, Inria, CNRS, UMR 7598 Laboratoire Jacques-Louis Lions, Paris, France*
[c] *Inria, Palaiseau, France*

Abstract

This paper extends the unfitted Nitsche-XFEM method of [*Comput. Methods Appl. Mech. Engrg.*, 301, 300–335, 2016] to three-dimensional fluid-structure interaction problems with immersed thin-walled elastic solids. The fluid and solid domains are discretized with unfitted unstructured meshes. Discrete weak and strong discontinuities are allowed in the fluid and the coupling is enforced consistently via a fluid-sided Nitsche's type mortaring with suitable stabilization for robustness. Integration over cut-elements is handled via an efficient and robust intersection and sub-tesselation algorithm. The method includes a new approach for the treatment of partially intersected fluid domains. Several numerical examples are presented and discussed, which illustrate the capabilities of the proposed method.

1. Introduction

The mechanical interaction of an incompressible viscous fluid with an immersed thin-walled structure appears in a wide variety of engineering fields and biosystems. The applications span from biomechanics of cells deformation, physiological flows, such as, heart valve dynamics, to aeroelasticity of parachutes and sailing boats (see, e.g., Liu and Liu (2006); Van Loon et al. (2005); Han and Peskin (2018); Nakata and Liu (2012); Weymouth et al. (2006); Takizawa and Tezduyar (2012)). One the fundamental difficulties that has to be faced in the approximation of these systems is that the coupling with the thin-walled solid introduces weak and strong discontinuities in the fluid velocity and pressure fields, respectively. The preservation of these properties at the discrete level is known to have major implications on the accuracy of the resulting numerical method. In particular, pressure discontinuities across the interface are essential to guarantee interfacial mass conservation, whereas discontinuities in the velocity gradient are a key ingredient in the optimality of the convergence order.

The development of fluid-structure interaction (FSI) numerical methods has been extensively investigated within the last decades, usually striving for superior accuracy, robustness and efficiency. Based on the discretization at the fluid-solid interface, approaches for solving FSI problems are often classified as fitted (conforming) and unfitted (non-conforming) mesh methods. In the boundary-fitted framework, the fluid and solid meshes are conforming at their interface, and the fluid problem is typically solved in a deforming mesh which follows the motion of the solid mesh. Note that this requires a geometrical representation of the interface within the fluid mesh, which facilitates the introduction of weak and strong discontinuities at the discrete level. Within the works involving fitted approaches we recall the studies based on the Arbitrary Lagrangian-Eulerian (ALE) formulation (see, e.g., Donea et al. (1982); Nomura and Hughes (1992); Formaggia and Nobile (1999); Stein et al. (2003); Takizawa et al. (2012); Landajuela et al. (2017)) and the unified continuum modeling for FSI (see Hoffman et al. (2011); Jansson et al. (2017); Liu and Marsden (2018)). Fitted mesh methods facilitate by design the discretization of the interface conditions. Furthermore, the computation of quantities of interest, such as wall shear stress, at the interface, is simple and accurate. However, when large interfacial deflections are involved (with potential contact between solids) these methods are often cumbersome; in fact, the fluid mesh can become very distorted unless re-meshing or topological mesh changes

*Corresponding author: Miguel A. Fernández
Email addresses: fgerosa@stanford.edu (Fannie M. Gerosa), daniele.corti@inria.fr (Daniele C. Corti), frederic.alauzet@inria.fr (Frédéric Alauzet), miguel.fernandez@inria.fr (Miguel A. Fernández)

are performed (see, e.g. Alauzet (2014); Takizawa et al. (2014)), which mitigates the computational benefits of the boundary fitted framework.

Under these circumstances, unfitted meshes-based methods are a favoured approach. In this class of methods, the fluid and solid meshes are non conforming at the interface, and the solid mesh is free to move over the background fluid mesh. The numerical approaches generally differ on how the fluid and solid approximations are built and on how the interface coupling conditions are treated at the discrete level. Among these approaches, we can mention the Immersed Boundary (IBM) and Immersed Finite Element (IFEM) methods (see, e.g., Peskin (2002); Zhang et al. (2004); Mittal and Iaccarino (2005); Boffi et al. (2011); Wang and Zhang (2013)), where the solid appears as an external body force in the fluid equations, and the fictitious domain methods (see, e.g., Glowinski et al. (1999); Baaijens (2001); De Hart et al. (2003); Astorino et al. (2009); Boffi et al. (2015); Kamensky et al. (2015); Boffi and Gastaldi (2017); Boilevin-Kayl et al. (2019a)), where the kinematic constraint is imposed via Lagrange multipliers. In general, these methods have the reputation of being inaccurate in space, due to the discrete treatment of the interface conditions and/or the fact that the fluid spatial discretization does not allow for discontinuities across the interface (see, e.g., Peskin and Printz (1993); Griffith (2012); Kamensky et al. (2015); Strychalski and Guy (2016); Boilevin-Kayl et al. (2019b); Casquero et al. (2021)). Mesh adaptivity is known to alleviate these issues (see, e.g., Hachem et al. (2013); Bergmann et al. (2022)), but it does not remove the problem. A natural approach to enhance interfacial mass conservation consists in considering global discontinuous pressure approximations with higher order polynomials for the velocities (see, e.g., Baaijens (2001); Boffi et al. (2012b,a, 2015)), or directly divergence free velocity approximations (see Casquero et al. (2021)). An alternative approach, based on more standard velocity/pressure approximation spaces, consists in boosting the incompressibility constraint via grad-div stabilization in the interfacial zone (see, e.g., Kamensky et al. (2015); Galvin et al. (2012); Casquero et al. (2017); Boilevin-Kayl et al. (2019b)), but this suffers from severe ill-conditioning issues. The extended-FEM (XFEM) method, which combines a cut-FEM approach with a local enrichment (see, e.g., Zilian and Legay (2008); Gerstenberger and Wall (2008); Sawada and Tezuka (2011)), overcomes these difficulties at the price of introducing additional unknowns (Lagrange multipliers) and potential ill-conditioning issues (lack of robustness with respect to the interface cuts). These difficulties are overcome by the Nitsche-XFEM method which combines overlapping meshes (strong and weak discontinuities) with a treatment *à la* Nitsche's of the interface coupling (Lagrange multipliers free) and suitable stabilization on the interfacial zone for robustness (see Alauzet et al. (2016); Burman and Fernández (2014), and also Zonca et al. (2018) for an extension to the case of the coupling with thick-walled solid models).

The superior accuracy of the Nitsche-FEM approach comes however at a price (see, e.g., Boilevin-Kayl et al. (2019b) for a comparison of some of the above mentioned methods), the method demands a much more involved computer implementation (e.g., with respect to traditional IBM or IFM methods) and requires a specific tracking of the mesh intersections. In Alauzet et al. (2016), only the two-dimensional (2D) case is investigated, with moderate interfacial displacements. In the present paper, we extend the unfitted Nitsche-XFEM method of Alauzet et al. (2016) to the three-dimensional (3D) setting, involving the coupling with non-linear shell models in large displacements. The mathematical formulation of the method is similar to the 2D case, however, the treatment of the *front* elements (partially intersected fluid domain) and the intersection of bulk-surface unstructured meshes is much more involved and requires specific methods. We hence propose a new approach to treat the *front* elements, the main idea consists in extending the solid mesh with a fictitious interface. Mesh intersection must be evaluated at each time iteration, so that the algorithm must be efficient and robust. Two main approaches have been proposed in the literature to perform the integration of the variational terms over the general polyhedra resulting from the intersection: sub-tessellation into triangular/tetrahedral elements and methods based on the Stokes theorem where a boundary representation of the bulk integrals is considered (e.g. Massing et al. (2013); Sudhakar et al. (2014); Antonietti et al. (2018); Chin and Sukumar (2020); Mayer et al. (2009, 2010); Shahmiri et al. (2011); Massing et al. (2013); Zonca et al. (2018)). In the former case, the majority of the studies rely on external meshing libraries. In the presented work, we propose an ad hoc three-dimensional intersection and sub-tetrahedralization/triangulation algorithm for general unstructured meshes cut by a triangular surface mesh. Extensive numerical evidence, in a series of three-dimensional examples involving large interfacial deflections, illustrates the performance and robustness of the proposed method.

The rest of the paper is organized as follows. Section 2 is devoted to the description of the continuous problem. In Section 3, we introduce the fully discrete formulation, for both fully and partially intersected fluid domains. Section 4 discusses important details of the mesh intersection and element duplication algorithms. In Section 5 is devoted to the numerical examples. Finally, Section 6 reports a summary of the conclusions and draws some perspectives of future work.

2. Problem setting

Let Ω be a given domain of $\mathbb{R}^3$, with boundary Γ. We consider a fluid-structure interaction system in which a thin-walled solid is immersed in a viscous incompressible fluid filling Ω. The reference configuration of the solid, represented by its mid-surface, is supposed to be given in terms of the oriented surface Σ, with unitary normal vector $\boldsymbol{n}$. The current position of the interface $\Sigma(t)$ is described in terms of a deformation map $\boldsymbol{\phi} : \Sigma \times \mathbb{R}^+ \longrightarrow \mathbb{R}^3$ as $\Sigma(t) = \boldsymbol{\phi}(\Sigma, t)$, with $\boldsymbol{\phi} \stackrel{\text{def}}{=} \boldsymbol{I}_\Sigma + \boldsymbol{d}$ and where $\boldsymbol{d}$ denotes the solid displacement. To ease the notation we set $\boldsymbol{\phi}_t \stackrel{\text{def}}{=} \boldsymbol{\phi}(\cdot, t)$, so that we also have $\Sigma(t) = \boldsymbol{\phi}_t(\Sigma)$. The corresponding fluid control volume $\Omega^{\text{f}}(t) \stackrel{\text{def}}{=} \Omega \backslash \Sigma(t) \subset \mathbb{R}^3$ is time-dependent, with its boundary partitioned as $\partial\Omega^{\text{f}}(t) = \Sigma(t) \cup \Gamma$. We will first assume that $\Sigma(t)$ separates Ω into two open domains $\Omega_1^{\text{f}}(t)$ and $\Omega_2^{\text{f}}(t)$ (see Figure 1). The general case of a partially intersected fluid domain will be discussed in Section 3.3. We denote the outward unit normal to $\Omega_i^{\text{f}}(t)$ on $\Sigma(t)$ by $\boldsymbol{n}_i, i = 1, 2$. Without loss of generality, we chose $\Omega_1(t)$ and $\Omega_2(t)$ so that $\boldsymbol{n}_1 = \boldsymbol{n}$ and $\boldsymbol{n}_2 = -\boldsymbol{n}$.

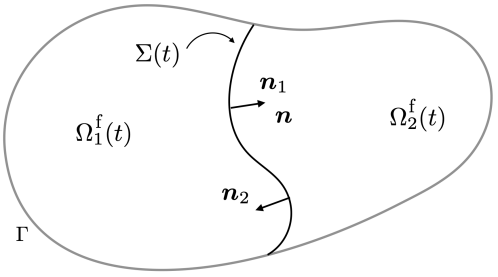


Figure 1: Geometric configuration of the fluid domain and of the immersed solid.

Some notation is needed to introduce quantities defined across the interface $\Sigma(t)$. For a given continuous scalar or tensorial field f defined in $\Omega^{\text{f}}(t)$, possibly discontinuous across $\Sigma(t)$, we define its sided-restrictions to $\Sigma(t)$, noted by f_1 and f_2, as

$$f_1(\boldsymbol{x}) \stackrel{\text{def}}{=} \lim_{\xi \to 0^-} f(\boldsymbol{x} + \xi \boldsymbol{n}_1), \quad f_2(\boldsymbol{x}) \stackrel{\text{def}}{=} \lim_{\xi \to 0^-} f(\boldsymbol{x} + \xi \boldsymbol{n}_2) \quad \forall \boldsymbol{x} \in \Sigma(t).$$

We also define the following jumps and average across the interface $\Sigma(t)$:

$$[\![f]\!] \stackrel{\text{def}}{=} f_1 - f_2, \quad [\![f\boldsymbol{n}]\!] \stackrel{\text{def}}{=} f_1 \boldsymbol{n}_1 + f_2 \boldsymbol{n}_2, \quad \{\!\{f\}\!\} \stackrel{\text{def}}{=} \frac{1}{2}(f_1 + f_2).$$

The considered coupled problem reads as follows: Find the fluid velocity and pressure $\boldsymbol{u} : \Omega \times \mathbb{R}^+ \to \mathbb{R}^3$, $p : \Omega \times \mathbb{R}^+ \to \mathbb{R}$, the solid displacement and velocity $\boldsymbol{d} : \Sigma \times \mathbb{R}^+ \to \mathbb{R}^3$, $\dot{\boldsymbol{d}} : \Sigma \times \mathbb{R}^+ \to \mathbb{R}^3$ such that, for all $t \in \mathbb{R}^+$, we have

$$\begin{cases} \rho^{\text{f}}(\partial_t \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u}) - \mathbf{div}\boldsymbol{\sigma}(\boldsymbol{u}, p) = \boldsymbol{0} & \text{in} \quad \Omega^{\text{f}}(t), \\ \text{div}\boldsymbol{u} = 0 & \text{in} \quad \Omega^{\text{f}}(t), \\ \boldsymbol{u} = \boldsymbol{0} & \text{on} \quad \Gamma, \end{cases} \tag{1}$$

$$\begin{cases} \rho^{\text{s}} \varepsilon \partial_t \dot{\boldsymbol{d}} + \boldsymbol{L}(\boldsymbol{d}) = \boldsymbol{T} & \text{on} \quad \Sigma, \\ \dot{\boldsymbol{d}} = \partial_t \boldsymbol{d} & \text{on} \quad \Sigma, \\ \boldsymbol{d} = \boldsymbol{0} & \text{on} \quad \partial\Sigma, \end{cases} \tag{2}$$

$$\begin{cases} \boldsymbol{\phi} = \boldsymbol{I}_\Sigma + \boldsymbol{d}, \quad \Sigma(t) = \boldsymbol{\phi}_t(\Sigma), \quad \Omega^{\text{f}}(t) = \Omega \backslash \Sigma(t), \\ \boldsymbol{u}_1 = \boldsymbol{u}_2 = \dot{\boldsymbol{d}} \circ \boldsymbol{\phi}_t^{-1} \quad \text{on} \quad \Sigma(t), \\ \displaystyle\int_\Sigma \boldsymbol{T} \cdot \boldsymbol{w} = -\int_{\Sigma(t)} [\![\boldsymbol{\sigma}(\boldsymbol{u}, p)\boldsymbol{n}]\!] \cdot \boldsymbol{w} \circ \boldsymbol{\phi}_t^{-1} \quad \forall \boldsymbol{w} \in \boldsymbol{W}. \end{cases} \tag{3}$$

complemented with standard initial conditions $\boldsymbol{u}(0) = \boldsymbol{u}_0, \boldsymbol{d}(0) = \boldsymbol{d}_0$ and $\dot{\boldsymbol{d}}(0) = \dot{\boldsymbol{d}}_0$. In the above system, the symbols ρ^{f}, ρ^{s} and ε respectively denote the fluid and solid densities and the thickness of the solid. Moreover, the standard notation is used for the fluid Cauchy stress tensor, namely,

$$\boldsymbol{\sigma}(\boldsymbol{u}, p) \stackrel{\text{def}}{=} 2\mu\boldsymbol{\epsilon}(\boldsymbol{u}) - p\boldsymbol{I}, \quad \boldsymbol{\epsilon}(\boldsymbol{u}) \stackrel{\text{def}}{=} \frac{1}{2}(\boldsymbol{\nabla u} + \boldsymbol{\nabla u}^{\mathrm{T}}),$$

where μ denotes the fluid dynamic viscosity. The abstract operator $\boldsymbol{L}$ in (2) represents the (non-linear) elastic operator of the solid. The relations in (3) respectively enforce the geometrical compatibility and the kinematic and dynamic coupling on the interface between the fluid and the solid media. In $(3)_3$, the symbol $\boldsymbol{W}$ denotes the space of admissible displacements in the solid and $\boldsymbol{T}$ the interfacial fluid stress exerted on the reference configuration.

Remark 1. Note that the fluid pressure in (1)-(3) is defined up to a constant.

3. Numerical method

This section is devoted to the numerical approximation of (1)-(3). For the time discretization, we consider a strongly coupled scheme with a explicit treatment of the interface location. An unfitted mesh approximation is considered in space, using overlapping meshes on the interface and a treatment *à la* Nitsche of the interface coupling (Nitsche-XFEM method, see Alauzet et al. (2016)).

3.1. Time discretization: strongly coupled scheme

In what follows, we will use the following notation for the first-order backward difference: $\partial_\tau x^n \stackrel{\text{def}}{=} (x^n - x^{n-1})/\tau$, where $\tau > 0$ denotes the time-step length. With the purpose of avoiding geometrical non-linearities in the fluid, we will discretize the geometric compatibility condition $(3)_1$ in an explicit fashion. For a given displacement approximation $\boldsymbol{d}^{n-1} \in \boldsymbol{W}$ at time level $n-1$, we define the deformation at time level n in an explicitly fashion as $\boldsymbol{\phi}^n \stackrel{\text{def}}{=} \boldsymbol{I}_\Sigma + \boldsymbol{d}^{n-1}$. This map characterizes the interface position at time level n, as $\Sigma^n \stackrel{\text{def}}{=} \boldsymbol{\phi}^n(\Sigma)$. We hence propose to update the physical fluid domain as

$$\Omega^{\mathrm{f},n} \stackrel{\text{def}}{=} \Omega \backslash \Sigma^n.$$

The resulting time semi-discrete scheme reads as follows, for $n \geq 1$:

1. Interface and fluid domain update:

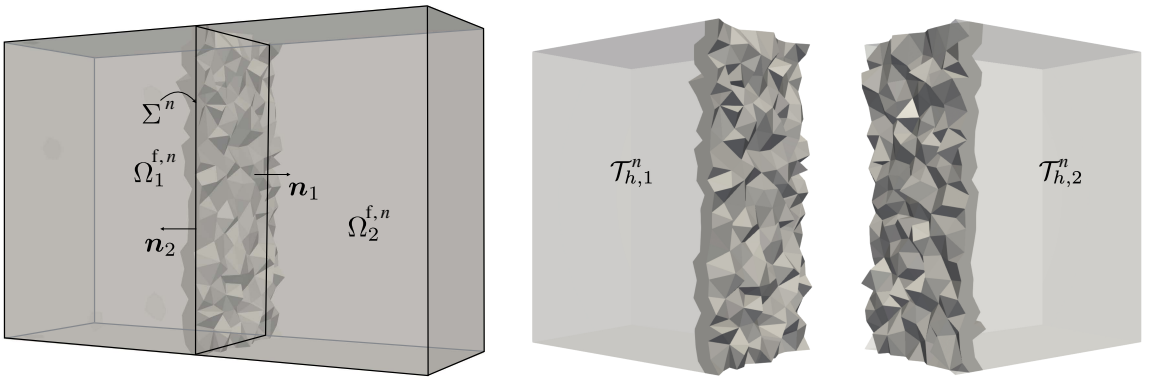
$$\boldsymbol{\phi}^n = \boldsymbol{I}_\Sigma + \boldsymbol{d}^{n-1}, \quad \Sigma^n = \boldsymbol{\phi}^n(\Sigma), \quad \Omega^{\mathrm{f},n} = \Omega \backslash \Sigma^n. \tag{4}$$

2. Fluid-solid coupled system:

$$\begin{cases} \rho^{\mathrm{f}}(\partial_\tau \boldsymbol{u}^n + \boldsymbol{u}^{n-1} \cdot \nabla \boldsymbol{u}^n) - \mathbf{div}\boldsymbol{\sigma}(\boldsymbol{u}^n, p^n) = \boldsymbol{0} & \text{in} \quad \Omega^{\mathrm{f},n}, \\ \mathrm{div}\boldsymbol{u}^n = 0 & \text{in} \quad \Omega^{\mathrm{f},n}, \\ \boldsymbol{u}^n = \boldsymbol{0} & \text{on} \quad \Gamma, \end{cases} \tag{5}$$

$$\begin{cases} \rho^{\mathrm{s}}\varepsilon\partial_\tau \dot{\boldsymbol{d}}^n + \boldsymbol{L}(\boldsymbol{d}^n) = \boldsymbol{T}^n & \text{on} \quad \Sigma, \\ \dot{\boldsymbol{d}} = \partial_t \boldsymbol{d} & \text{on} \quad \Sigma, \\ \boldsymbol{d} = \boldsymbol{0} & \text{on} \quad \partial\Sigma, \end{cases} \tag{6}$$

$$\begin{cases} \boldsymbol{u}^n = \dot{\boldsymbol{d}}^n \circ (\boldsymbol{\phi}^n)^{-1} & \text{on} \quad \Sigma^n, \\ \displaystyle\int_\Sigma \boldsymbol{T}^n \cdot \boldsymbol{w} = -\int_{\Sigma^n} [\![\boldsymbol{\sigma}(\boldsymbol{u}^n, p^n)\boldsymbol{n}]\!] \cdot \boldsymbol{w} \circ (\boldsymbol{\phi}^n)^{-1} & \forall \boldsymbol{w} \in \boldsymbol{W}. \end{cases} \tag{7}$$



(a) (b)

Figure 2: (a) Three-dimensional unfitted fluid-solid meshes (with the intersected fluid elements highlighted). (b) The triangulations $\mathcal{T}_{h,1}^n$ and $\mathcal{T}_{h,2}^n$ with the overlapping region in dark grey.

3.2. Unfitted mesh approximation: fully discrete scheme

We propose to approximate the fluid velocity and pressure with triangulations of $\Omega^{\mathrm{f},n}$ which are independent of the interface Σ^n. The strong and weak interfacial discontinuities are included in the discrete fluid solution my means of overlapping meshes (see Hansbo and Hansbo (2004); Becker et al. (2009); Alauzet et al. (2016)), which are build by duplicating the fluid elements that are intersected by the interface Σ^n, as shown in Figure 2. In this framework we enforce the coupling conditions (7) by Nitsche's type mortaring as reported in Burman and Fernández (2014); Alauzet et al. (2016); Hansbo et al. (2004).

For sake of simplicity, we assume that both $\Omega^{\mathrm{f},n}$ and Σ^n are polyhedral. Let $\{\mathcal{T}_h^{\mathrm{s}}\}_{0<h<1}$ be a family of quasi-uniform triangulations of Σ and we consider the standard space of continuous piecewise affine functions:

$$X_h^{\mathrm{s}} \stackrel{\mathrm{def}}{=} \left\{ v_h \in C^0(\overline{\Sigma}) \,\middle|\, v_{h|K} \in \mathbb{P}_1(K), \quad \forall K \in \mathcal{T}_h^{\mathrm{s}} \right\}.$$

The discrete space for the solid displacement and velocity approximations is hence defined as $\boldsymbol{W}_h = [X_h^{\mathrm{s}}]^3 \cap \boldsymbol{W}$, where $\boldsymbol{W} = [H^1(\Sigma)]_0^d$ denotes the space of admissible displacements. The weak form of the abstract solid elastic operator $\boldsymbol{L}$ in (2) is assumed to be given by a form $a^{\mathrm{s}} : \boldsymbol{W} \times \boldsymbol{W} \longrightarrow \mathbb{R}$, linear with respect of its second argument.

For the fluid, we introduce two families of quasi-uniform tetrahedral meshes $\{\mathcal{T}_{h,i}^n\}_{0<h<1}$, $i = 1, 2$, where each $\mathcal{T}_{h,i}^n$ covers the i-th fluid region $\Omega_i^{\mathrm{f},n}$ separated by Σ^n. Each mesh $\mathcal{T}_{h,i}^n$ is fitted to the exterior boundary Γ_i but not to Σ^n. Moreover, we assume that for every element $K \in \mathcal{T}_{h,1}^n \cap \mathcal{T}_{h,2}^n$ we have $K \cap \Sigma^n \neq \emptyset$. We denote by $\Omega_{h,i}^n$ the domain covered by $\mathcal{T}_{h,i}^n$, namely,

$$\Omega_{h,i}^n \stackrel{\mathrm{def}}{=} \mathrm{int}\left(\cup_{K \in \mathcal{T}_{h,i}^n} K \right).$$

Note that the mesh composed by $\mathcal{T}_{h,1}^n \cup \mathcal{T}_{h,2}^n$ is a conforming mesh of the whole fluid domain Ω^{f}, but which does not contains any geometrical representation of the interface Σ^n. For $i = 1, 2$, we can hence introduce the following spaces of continuous piecewise affine functions:

$$X_{h,i}^n \stackrel{\mathrm{def}}{=} \left\{ v_h \in C^0(\overline{\Omega_{h,i}^n}) : v_{h|K} \in \mathbb{P}_1(K), \quad \forall K \in \mathcal{T}_{h,i}^n \right\},$$

Associated with $X_{h,i}^n$ we define the following spaces

$$\boldsymbol{V}_{h,i}^n \stackrel{\mathrm{def}}{=} [X_{h,i}^n]^3 \cap [H_\Gamma^1(\Omega^{\mathrm{f}})]^3, \quad Q_{h,i}^n \stackrel{\mathrm{def}}{=} X_{h,i}^n.$$

For the approximation of the fluid velocity and pressure we consider the following discrete product spaces

$$\boldsymbol{V}_h^n \stackrel{\mathrm{def}}{=} \boldsymbol{V}_{h,1}^n \times \boldsymbol{V}_{h,2}^n, \quad Q_h^n \stackrel{\mathrm{def}}{=} (Q_{h,1}^n \times Q_{h,2}^n) \cap L_0^2(\Omega^{\mathrm{f},n}). \tag{8}$$

Note that, owing to the overlap between $\Omega_{h,1}^n$ and $\Omega_{h,2}^n$, interfacial strong and weak discontinuities are included in these discrete approximation spaces (see Figure 3 for an illustration in 1D).

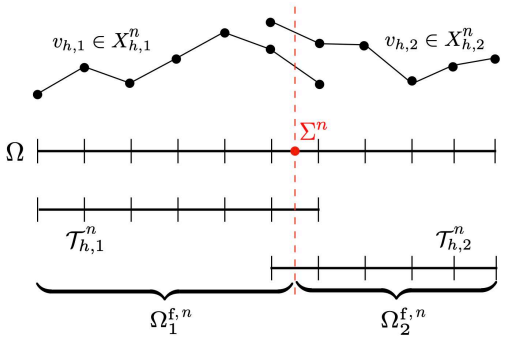


Figure 3: 1D illustration of the discrete spaces $X^n_{h,i}$ supported by the overlapping meshes $\mathcal{T}^n_{h,i}$, $i = 1, 2$.

For the spatial approximation of the fluid, we introduce the following discrete bi-linear and tri-linear forms respectively associated to the Stokes and convective operators in (5):

$$\begin{aligned} a_{\Omega^{\mathrm{f},n}}(\,(\boldsymbol{u}_h, p_h)\,,(\boldsymbol{v}_h, q_h)\,) &\stackrel{\text{def}}{=} 2\mu(\boldsymbol{\epsilon}(\boldsymbol{u}_h), \boldsymbol{\epsilon}(\boldsymbol{v}_h))_{\Omega^{\mathrm{f},n}} - (p_h, \mathrm{div}\boldsymbol{v}_h)_{\Omega^{\mathrm{f},n}} + (q_h, \mathrm{div}\boldsymbol{u}_h)_{\Omega^{\mathrm{f},n}}, \\ c^n(\boldsymbol{z}_h; \boldsymbol{u}_h, \boldsymbol{v}_h) &\stackrel{\text{def}}{=} \rho^{\mathrm{f}}(\boldsymbol{z}_h \cdot \boldsymbol{\nabla}\boldsymbol{u}_h, \boldsymbol{v}_h)_{\Omega^{\mathrm{f},n}} + \frac{\rho^{\mathrm{f}}}{2}((\mathrm{div}\boldsymbol{z}_h)\boldsymbol{u}_h, \boldsymbol{v}_h)_{\Omega^{\mathrm{f},n}} \\ &\quad - \rho^{\mathrm{f}}(\{\!\{\boldsymbol{z}_h\}\!\} \cdot \boldsymbol{n}[\![\boldsymbol{u}_h]\!], \{\!\{\boldsymbol{v}_h\}\!\})_{\Sigma^n} - \frac{\rho^{\mathrm{f}}}{2}([\![\boldsymbol{z}_h \cdot \boldsymbol{n}]\!], \{\!\{\boldsymbol{u}_h \cdot \boldsymbol{v}_h\}\!\})_{\Sigma^n}. \end{aligned} \tag{9}$$

The three last there terms are added to guarantee that $c^n(\boldsymbol{v}_h; \boldsymbol{z}_h, \boldsymbol{z}_h) = 0$ for all $\boldsymbol{z}_h \in \boldsymbol{V}^n_h$, as in standard DG methods (see, e.g., Di Pietro and Ern (2012)).

In order to cope with the lack of inf−sup stability of the discrete spaces (8) and with the numerical instabilities for large local Reynolds number, the continuous interior penalty stabilization method (CIP) will be used. The associated symmetric velocity and pressure stabilization operators are given by (see, e.g., Burman et al. (2006); Burman and Fernández (2007)):

$$\begin{aligned} s^n_{\mathrm{v},h}(\boldsymbol{z}_h; \boldsymbol{u}_h, \boldsymbol{v}_h) &\stackrel{\text{def}}{=} \gamma_{\mathrm{v}} h^2 \sum_{i=1}^{2} \sum_{F \in \mathcal{F}^n_{h,i}} \xi(\mathrm{Re}_F(\boldsymbol{z}_h)) \|\boldsymbol{z}_h \cdot \boldsymbol{n}\|_{L^\infty(F)} ([\![\boldsymbol{\nabla}\boldsymbol{u}_h]\!]_F, [\![\boldsymbol{\nabla}\boldsymbol{v}_h]\!]_F)_F, \\ s^n_{\mathrm{p},h}(\boldsymbol{z}_h; p_h, q_h) &\stackrel{\text{def}}{=} \gamma_{\mathrm{p}} h^2 \sum_{i=1}^{2} \sum_{F \in \mathcal{F}^n_{h,i}} \frac{\xi(\mathrm{Re}_F(\boldsymbol{z}_h))}{\|\boldsymbol{z}_h\|_{L^\infty(F)}} ([\![\boldsymbol{\nabla} p_h]\!]_F, [\![\boldsymbol{\nabla} q_h]\!]_F)_F, \end{aligned}$$

where $\mathcal{F}^n_{h,i}$ denotes the set of interior faces of $\mathcal{T}^n_{h,i}$, $\mathrm{Re}_F(\boldsymbol{z}_h) \stackrel{\text{def}}{=} \rho^{\mathrm{f}} \|\boldsymbol{z}_h\|_{L^\infty(F)} h \mu^{-1}$ denotes the local Reynolds number, $\xi(x) \stackrel{\text{def}}{=} \min\{1, x\}$ is a cut-off function and $\gamma_{\mathrm{p}}, \gamma_{\mathrm{v}} > 0$ are user-defined parameters. Furthermore, in order to guarantee robustness with respect to the way the interface Σ^n is cutting the overlapping meshes $\mathcal{T}^n_{h,i}$, we introduce the time-dependent ghost-penalty operator, given by (see, e.g., Burman (2010)):

$$g^n_h(\boldsymbol{u}_h, \boldsymbol{v}_h) \stackrel{\text{def}}{=} \gamma_{\mathrm{g}} \mu h \sum_{i=1}^{2} \sum_{F \in \mathcal{F}^{\Sigma^n}_{h,i}} ([\![\boldsymbol{\nabla}\boldsymbol{u}_{h,i}]\!]_F, [\![\boldsymbol{\nabla}\boldsymbol{v}_{h,i}]\!]_F)_F,$$

where $\mathcal{F}^{\Sigma^n}_{h,i}$ denotes the set of interior faces of the elements of $\mathcal{T}^n_{h,i}$ intersected by Σ^n. Finally, we collect all the above terms in a single fluid contribution

$$\begin{aligned} a^{\mathrm{f}}_{\Omega^{\mathrm{f},n},h}(\boldsymbol{z}_h; (\boldsymbol{u}_h, p_h)\,, (\boldsymbol{v}_h, q_h)\,) &\stackrel{\text{def}}{=} c^n\,(\boldsymbol{z}_h, \boldsymbol{u}_h, \boldsymbol{v}_h) + a_{a_{\Omega^{\mathrm{f},n}}}(\,(\boldsymbol{u}_h, p_h)\,, (\boldsymbol{v}_h, q_h)\,) \\ &\quad + s^n_{\mathrm{v},h}\,(\boldsymbol{z}_h; \boldsymbol{u}_h, \boldsymbol{v}_h) + s^n_{\mathrm{p},h}\,(\boldsymbol{z}_h; p_h, q_h) + g^n_h\,(\boldsymbol{u}_h, \boldsymbol{v}_h)\,. \end{aligned}$$

By combining the explicit treatment of the geometric compatibility (4) with a Nitsche fluid-sided mortaring of the the strong coupling (5)-(7), we get the fully discrete solution procedure reported in Algorithm 1.

Algorithm 1: Nitsche-XFEM strongly coupled scheme.

For $n \geq 1$:

1. Interface update:
$$\boldsymbol{\phi}_h^n = \boldsymbol{I}_\Sigma + \boldsymbol{d}_h^{n-1}, \quad \Sigma^n = \boldsymbol{\phi}_h^n(\Sigma), \quad \Omega^{\mathrm{f},n} = \Omega \backslash \Sigma^n.$$

2. Find $(\boldsymbol{u}_h^n, p_h^n, \dot{\boldsymbol{d}}_h^n, \boldsymbol{d}_h^n) \in \boldsymbol{V}_h^n \times Q_h^n \times \boldsymbol{W}_h \times \boldsymbol{W}_h$, such that $\dot{\boldsymbol{d}}_h^n = \partial_\tau \boldsymbol{d}_h^n$ and

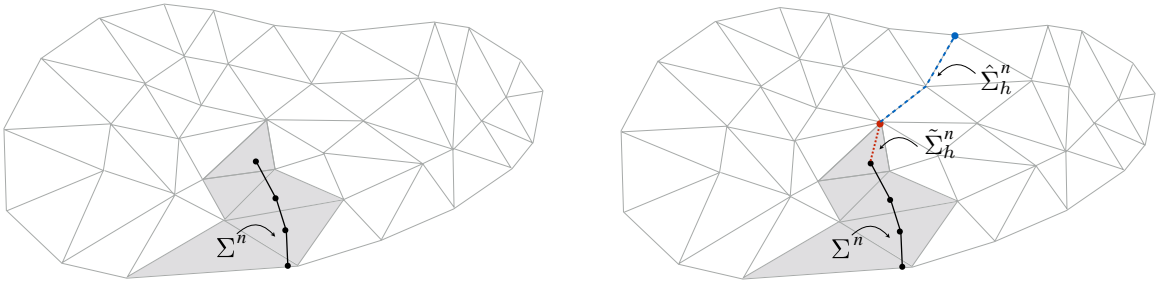
$$\begin{aligned} &\rho^{\mathrm{f}}(\partial_\tau \boldsymbol{u}_h^n, \boldsymbol{v}_h)_{\Omega^{\mathrm{f},n}} + a^{\mathrm{f}}_{\Omega^{\mathrm{f},n},h}(\boldsymbol{u}_h^n; (\boldsymbol{u}_h^n, p_h^n), (\boldsymbol{v}_h, q_h)) + \rho^{\mathrm{s}} \varepsilon(\partial_\tau \dot{\boldsymbol{d}}_h^n, \boldsymbol{w}_h)_\Sigma + a^{\mathrm{s}}(\boldsymbol{d}_h^n; \boldsymbol{w}_h) \\ &- \sum_{i=1}^{2} (\boldsymbol{\sigma}(\boldsymbol{u}_{h,i}^n, p_{h,i}^n)\boldsymbol{n}_i, \boldsymbol{v}_{h,i} - \boldsymbol{w}_h)_{\Sigma^n} - \sum_{i=1}^{2} (\boldsymbol{u}_{h,i}^n - \dot{\boldsymbol{d}}_h^n, \boldsymbol{\sigma}(\boldsymbol{v}_{h,i}, -q_{h,i}))\boldsymbol{n}_i)_{\Sigma^n} + \frac{\gamma\mu}{h} \sum_{i=1}^{2} (\boldsymbol{u}_{h,i}^n - \dot{\boldsymbol{d}}_h^n, \boldsymbol{v}_{h,i} - \boldsymbol{w}_h)_{\Sigma^n} = 0 \end{aligned} \tag{10}$$

for all $(\boldsymbol{v}_h, q_h, \boldsymbol{w}_h) \in \boldsymbol{V}_h^n \times Q_h^n \times \boldsymbol{W}_h$.

Due to the dynamic nature of problem, Algorithm 1 involves integrals of functions associated to different time levels, namely, e.g. $(\boldsymbol{u}_h^{n-1}, \widetilde{\boldsymbol{v}}_h)_{\Omega^{\mathrm{f},n}}$. At each iteration a new location of the structure is computed, together with a new sub-tetrahedralization. This requires the integration of products of functions that might be discontinuous at different locations in the same element. In order avoid the simultaneous intersection of different interface locations within the same fluid element, we consider the approach introduced in Alauzet et al. (2016); Fries and Zilian (2009), which basically consists in locally shifting the discontinuity at time $t^\star$ to the structure location at time t^n, where $t^\star$ refers to t^n and t^{n-1} respectively. In the case where the discontinuities are located in different elements, the quadrature is performed in a standard fashion since we keep track of the (previous) intersections at different times and we can treat each discontinuity separately.

3.3. Partially intersected fluid domain

In this section, we propose an extension of Algorithm 1 to the case in which the interface Σ^n only partially intersects the domain Ω. We introduce a procedure whose purpose is twofold: first, connect the duplicated fluid elements intersected by the solid boundary elements with the rest of the fluid domain; and secondly, facilitate the definition of the fluid discrete spaces. For the sake of comparison, we briefly recall the two-dimensional procedure proposed in Alauzet et al. (2016). In 2D, we refer to the boundary solid vertex as *tip*, while in 3D the immersed boundary is identified as a *front*. Additionally, we will name with *fluid tip elements* the fluid simplexes that are intersected by the solid boundary *tip* in 2D or the boundary *front* in 3D. The strategy proposed in Alauzet et al. (2016)



(a) Initial fluid and solid triangulations.

(b) Virtual interface extension $\Sigma_h^{\mathrm{tip},n} \overset{\mathrm{def}}{=} \widetilde{\Sigma}_h^n \cup \widehat{\Sigma}_h^n$.

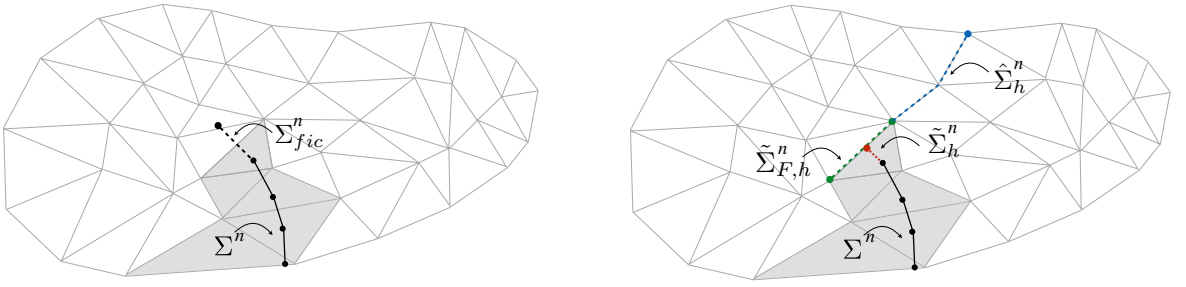
Figure 4: Treatment of the tip elements in 2D from Alauzet et al. (2016).

for the 2D case consists in introducing a virtual interface $\widetilde{\Sigma}_h^n$ which closes the fluid domain within the cut element. The resulting fluid-fluid fictitious prolongation (see Figure 4) is defined in terms of the partition $\Sigma_h^{\mathrm{tip},n} \overset{\mathrm{def}}{=} \widetilde{\Sigma}_h^n \cup \widehat{\Sigma}_h^n$, where:

- $\widetilde{\Sigma}_h^n$ is the prolongation of the interface tip up to the fluid vertex which is opposite to the edge intersected by Σ^n;

- $\widehat{\Sigma}_h^n$ is arbitrary chosen, but aligned with the edges of the fluid mesh. It connects the fluid tip vertex with the fluid external boundary Γ.

We recall that $\widehat{\Sigma}_h^n$ is only used as mathematical object to define the discrete spaces (i.e., it is never constructed in practice). Unfortunately, the extension of this approach to 3D is not straightforward. Particularly, there is no a clear definition of the 3D counterpart of $\widetilde{\Sigma}_h^n$.



(a) Initial fluid and solid triangulation with the fictitious interface Σ_{fic}^n.

(b) Virtual interface prolongation $\Sigma_h^{\text{tip},n} = \widetilde{\Sigma}_h^n \cup \widetilde{\Sigma}_{F,h}^n \cup \widehat{\Sigma}_h^n$.

Figure 5: New treatment of the tip elements in 2D.

We propose a new approach which avoids the need of a closure strategy in the fluid tip elements. The main idea consists in introducing a fictitious prolongation Σ_{fic}^n of the interface Σ^n (see Figure 5(a), and Section 3.3.1 below for the details on the construction of this extension). As shown in Figure 5 for the 2D case, the fluid-fluid fictitious prolongation $\Sigma_h^{\text{tip},n}$ is now defined as $\Sigma_h^{\text{tip},n} = \widetilde{\Sigma}_h^n \cup \widetilde{\Sigma}_{F,h}^n \cup \widehat{\Sigma}_h^n$, where:

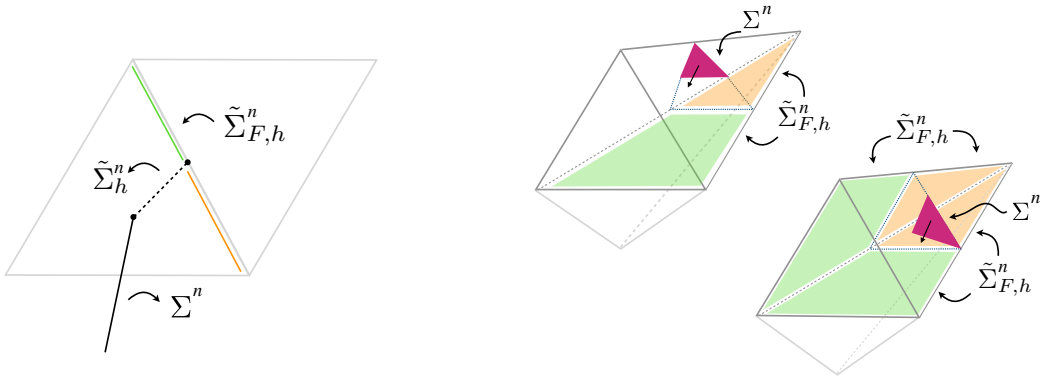
- $\widetilde{\Sigma}_h^n$ represents the intersection of the fictitious interface Σ_{fic}^n that belongs to the *fluid tip simplex*. Note that, contrarily to the approach proposed in Alauzet et al. (2016), the configuration of $\widetilde{\Sigma}_h$ is now arbitrary with respect to the fluid element;
- $\widetilde{\Sigma}_{F,h}^n$ stands for the edge or faces of the fluid tip element that are intersected by the fictitious interface Σ_{fic}^n. It should be observed that in two dimensions there is only one *tip edge*, while in three dimensions multiple faces can be considered as *tip faces* (see Figure 6);
- $\widehat{\Sigma}_h^n$, similarly to Alauzet et al. (2016), is arbitrary chosen, aligned with the edges or faces of the fluid mesh and, additionally, it connects the fluid tip edge/faces with the fluid boundary Γ.

We now proceed, as in Section 3.2, by introducing two overlapping meshes $\mathcal{T}_{h,i}^{\text{f},n}$, $i = 1, 2$. The overlapping region still reduces to the set of elements intersected by Σ^n, since the fluid elements intersected only by the fictitious interface Σ_{fic}^n are not duplicated (see Figure 7). The associated discrete spaces $\boldsymbol{V}_{h,i}^n$ and $Q_{h,i}^n$, are then defined as in Section 3.2. The fluid approximation spaces are hence defined as:

$$\boldsymbol{V}_h^n \overset{\text{def}}{=} \left\{ \boldsymbol{v}_h = (\boldsymbol{v}_{h,1}, \boldsymbol{v}_{h,2}) \in \boldsymbol{V}_{h,1}^n \times \boldsymbol{V}_{h,2}^n : \boldsymbol{v}_{h,1} = \boldsymbol{v}_{h,2} \text{ on } \widehat{\Sigma}_h^n \right\},$$
$$Q_h^n \overset{\text{def}}{=} \left\{ q_h = (q_{h,1}, q_{h,2}) \in Q_{h,1}^n \times Q_{h,2}^n : q_{h,1} = q_{h,2} \text{ on } \widehat{\Sigma}_h^n \right\} \cap L_0^2(\Omega^{\text{f},n}).$$

This new approach results in functions that are continuous in $\Omega^{\text{f},n} \backslash (\Sigma^n \cup \widetilde{\Sigma}_h^n \cup \widetilde{\Sigma}_{F,h}^n)$ and discontinuous across $\Sigma^n \cup \widetilde{\Sigma}_h^n \cup \widetilde{\Sigma}_{F,h}^n$. The kinematic and dynamic coupling across the fluid-fluid fictitious interfaces $\widetilde{\Sigma}_h^n \cup \widetilde{\Sigma}_{F,h}^n$ is enforced by adding into (10) the following standard DG terms (see, e.g., Di Pietro and Ern (2012)):

$$-\rho^{\text{f}}(\{\!\{\boldsymbol{z}_h\}\!\} \cdot \boldsymbol{n}[\![\boldsymbol{u}_h]\!], \{\!\{\boldsymbol{v}_h\}\!\})_{\widetilde{\Sigma}_h^n \cup \widetilde{\Sigma}_{F,h}^n} - \frac{\rho^{\text{f}}}{2}([\![\boldsymbol{z}_h \cdot \boldsymbol{n}]\!], \{\!\{\boldsymbol{u}_h \cdot \boldsymbol{v}_h\}\!\})_{\widetilde{\Sigma}_h^n \cup \widetilde{\Sigma}_{F,h}^n}$$
$$- (\{\!\{\boldsymbol{\sigma}(\boldsymbol{u}_h^n, p_h^n)\}\!\}\boldsymbol{n}, [\![\boldsymbol{v}_h]\!])_{\widetilde{\Sigma}_h^n \cup \widetilde{\Sigma}_{F,h}^n} - (\{\!\{\boldsymbol{\sigma}(\boldsymbol{v}_h, -q_h)\}\!\}\boldsymbol{n}, [\![\boldsymbol{u}_h^n]\!])_{\widetilde{\Sigma}_h^n \cup \widetilde{\Sigma}_{F,h}^n} + \frac{\gamma\mu}{h}([\![\boldsymbol{u}_h^n]\!], [\![\boldsymbol{v}_h]\!])_{\widetilde{\Sigma}_h^n \cup \widetilde{\Sigma}_{F,h}^n}. \quad (11)$$



(a) In 2D: tip edges.

(b) In 3D: tip faces.

Figure 6: Edge or faces of the fluid tip element that are intersected by the fictitious interface Σ^n_{fic}.

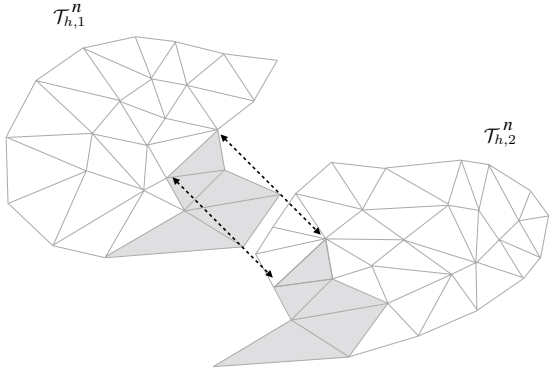


Figure 7: Overlapping triangulations in 2D with partially intersected fluid domain.

Remark 2. It should be noted that the terms (11) act on the whole fluid-fluid fictitious interface $\widetilde{\Sigma}^n_h \cup \widetilde{\Sigma}^n_{F,h}$. The jumps and averages across the tip faces $\widetilde{\Sigma}^n_{F,h}$ must be interpreted between the physical region of the tip fluid element and its respective opposite element in the fluid triangulation (see Figure 6).

3.3.1. Example of fictitous interface Σ^n_{fic}

The choice of physical fictitous interface Σ^n_{fic} is arbitrary. We discuss here a simple strategy to build Σ^n_{fic}, which basically consists in (see Figure 8):

1. Prolongating the reference solid domain Σ by Σ_{fic}, so that the new solid computational domain is $\overline{\Sigma \cup \Sigma_{\text{fic}}}$;
2. Modifying the solid contributions in (10) so that the fictitious solid region Σ_{fic} is mechanically slave of the physical solid interface Σ by enforcing (only) the kinematic continuity at the interface between the two regions.

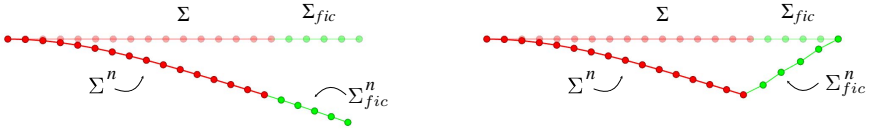
In practice, the uncoupling between the reference physical Σ and fictitious Σ_{fic} solid regions is performed at the algebraic level of the solid solver, during the elementary evaluations of the residual and tangent matrices. We provide here some details in the linear case. The finite element system over the whole solid computational domain has the following structure:

$$\begin{bmatrix} \mathrm{A_{PP}} & \mathrm{A_{PI}} & 0 \\ \mathrm{A_{IP}} & \mathrm{A_{II}} & \mathrm{A_{IF}} \\ 0 & \mathrm{A_{FI}} & \mathrm{A_{FF}} \end{bmatrix} \begin{bmatrix} \mathrm{x_P} \\ \mathrm{x_I} \\ \mathrm{x_F} \end{bmatrix} = \begin{bmatrix} \mathrm{b_P} \\ \mathrm{b_I} \\ \mathrm{b_F} \end{bmatrix}, \tag{12}$$

where, the arrays x_P, x_F, x_I respectively denote the solid degrees of freedom in the physical region, fictitious region and interface between the two regions. The right-hand side arrays b_P, b_I and b_F stands for the algebraic counterpart of body forces and/or time-stepping terms. The fundamental idea consists in removing the coupling block A_{IF} in (12), which yields

$$\begin{bmatrix} A_{PP} & A_{PI} & 0 \\ A_{IP} & \widetilde{A}_{II} & 0 \\ 0 & A_{FI} & A_{FF} \end{bmatrix} \begin{bmatrix} x_P \\ x_I \\ x_F \end{bmatrix} = \begin{bmatrix} b_P \\ b_I \\ \widetilde{b}_F \end{bmatrix}, \tag{13}$$

where the fictitious right-hand side $\widetilde{b}_F$ does not contain body forces related to the physical region anymore, but only time-stepping terms, and $\widetilde{A}_{II}$ does not include the contributions from elements belonging to the non-physical side. Note that, in this modified system, the physical degrees of freedom x_P and x_I are uncoupled from the fictitious degrees of freedom x_F. Indeed, the first two rows of (13) correspond to a solid problem in the physical region with homogeneous Neumann boundary conditions on the interface between the two regions. The last row of (13) is nothing but a solid problem in the fictitious region with Dirichlet conditions on the interface between the two regions (note that the prescribed Dirichlet data x_I is known from the first two rows of (13)). An illustration of this is given in Figure 8, for different external boundary conditions on the fictitious region. In this example, we impose a vertical volumetric force on the physical region Σ and different boundary conditions on the right extremity of Σ_{fic}, zero traction in 8(a) and zero displacement in 8(b). The displacement of Σ is clearly independent of the displacement of Σ_{fic}.



(a) Zero traction on the right extremity.

(b) Zero displacement on the right extremity.

Figure 8: Illustration of the slave nature of Σ^n_{fic} with respect to Σ^n. The solid problem is solved in Σ_{fic} with different boundary conditions on its right extremity.

Remark 3. Note that the physical interface Σ^n is the sole part of the deformed solid mesh which intervenes in the solid approximation and fluid-structure interaction coupling.

4. Interface tracking and overlapping meshes

In this section, we discuss how the issues related to mesh intersection (§4.1) and element duplication (§4.2) are handled in the presented work.

4.1. Intersection algorithm

As pointed out above, the Nitsche-XFEM method requires a specific track of the interface inside the fluid mesh, and the construction of a sub-tesselation of the cut elements, with the sole purpose of numerical integration. The development of a robust and efficient mesh intersection algorithm is an essential ingredient for the implementation of Algorithm 1. This is a problem far from being trivial in 3D with arbitrary unstructured meshes. It should be noted that there is no a priori information of which fluid element is intersected by the solid mesh and vice versa. Furthermore, for complex polyhedra, a sub-tetrahedralization cannot be guaranteed in 3D without introducing additional vertices. Most of the methods proposed in the literature often relay on external meshing libraries for the local subtetrahedralization. In the following paragraphs, we introduce a 3D intersection algorithm between volume (tetrahedral) and surface (triangular) unstructured meshes, including subtetrahedralization and subtriangulation of the cut fluid and solid elements, without resorting to black-box meshing libraries.

Let consider a domain Ω and its triangulation $\mathcal{T}_h$ of simplexes K, such that

$$\overline{\Omega} = \bigcup_{K \in \mathcal{T}_h} K.$$

The triangulation is called *conforming*, if the non empty intersection F between two elements K_1, K_2, namely $F = K_1 \cap K_2 \neq \emptyset$ with $K_1, K_2 \in \mathcal{T}_h$ and $K_1 \neq K_2$, is either a whole face, edge or a vertex of the triangulation. If this condition is not fulfilled the triangulation is called *non-conforming*. An important feature of intersection algorithms is the notion of *geometric tolerance*. Given two geometric objects K_1 and K_2, we say that K_1 intersects K_2 and *vice versa*, if the minimum distance between K_1 and K_2 is smaller than the *geometric tolerance*. In addition, the *tolerance* is also used to merge or collapse two different geometric elements (i.e., if the distance between two nodes is smaller that the *tolerance*, they are merged into the same node). In the context of *conforming* meshes the notion of *tolerance* is global (unique for the whole mesh), while for *non-conforming* meshes the *tolerance* can be defined locally (it can be adapted on each element, i.e., we can have different tolerance in each tetrahedron). The 2D intersection and subtriangulation

Algorithm 2: 3D-2D intersection algorithm.

```
/* Localization, stage I.1                                              */
forall V ∈ T_h^s do
    M(V) ← Localize( T_h^f, V)                          // Initialize M
/* Initialization, stage I.2                                            */
forall T ∈ T_h^s \ {T ∈ Σ_fic} do
    forall V ∈ T do
        if M(V) ≠ ∅ then
            K ← M(V)
            if K ∉ L then
                L ← L ∪ {K}                              // Initialize L
            if T ∉ L_K then
                L_K ← L_K ∪ {T}                          // Initialize L_K
/* Intersection, stage I.3                                              */
repeat
    forall K ∈ L do
        forall T ∈ L_K not yet intersected do
            VerticesInsertion( T_K^f, T_K^s, K, T)       // Stage I.3.1
            EdgesInsertion( T_K^f, T_K^s, K, T)          // Stage I.3.2
            FaceInsertion( T_K^f, T_K^s, K, T)           // Stage I.3.3
            Update( L_K)
        Update( L)
until ∃K ∈ L : ∃T ∈ L_K not yet intersected
/* Consistency checks, stage I.4                                        */
Check()
/* Post-processing, stage I.5                                           */
forall K ∈ L do
    T_Int^f ← T_Int^f ∪ T_K^f                            // Assembling T_Int^f
    T_Int^s ← T_Int^s ∪ T_K^s                            // Assembling T_Int^s
Labeling()                                       // Label LEFT and RIGHT
```

algorithm presented in Alauzet et al. (2016) is based on a *conformal* approach. Indeed, the solid mesh vertices and edges are inserted directly into the fluid mesh, and every intersection detected results in a conformal splitting of the elements involved. Numerical experiments (even in 2D) show lack of robustness when the discretization parameter h_s of the solid mesh is considered significantly smaller than the one of the fluid mesh h_f. In 3D, invalid configurations (e.g., tetrahedra generated from coplanar vertices) already appear whenever $h_s \approx h_f$. Adjusting (both increasing and reducing) the geometric tolerance mitigates the problem, but doing so globally is unfeasible in practice, due to a conformity constraint, since it requires restarting the intersection process each time the *tolerance* value changes.

The algorithm presented thereafter overcomes these issues by computing the intersection between each fluid el-

ement and the solid mesh separately and independently, i.e. treating locally each fluid element and by ignoring any connection to its neighbours. A local notion of *tolerance* is considered in order to avoid invalid configurations. The resulting intersected meshes are hence *non-conforming* in general. Before providing a detailed description of the algorithm, it is worth recalling that the quality of the intersected mesh is not a concern here, since the sole purpose is numerical quadrature in cut-elements. Therefore, only mesh validity has to be checked.

We start by introducing some notation which will be widely used in the following. Let $\mathcal{T}_h^{\mathrm{s}}$ and $\mathcal{T}_h^{\mathrm{f}}$ be the solid and fluid meshes, respectively. Given a general mesh $\mathcal{T}$, we denote by $\mathcal{V}_{\mathcal{T}}$ the set of vertices of $\mathcal{T}$ and by $\mathcal{K}_{\mathcal{T}}$ the set of elements of $\mathcal{T}$. We denote with V, E, F, T and K a generic vertex, edge, face, triangle and tetrahedron, respectively. Moreover, we define the following sets (with a little abuse of notation):

- $\mathcal{B}(V) \stackrel{\text{def}}{=} \{K \in \mathcal{T} : V \in K\}$ denotes the set of tetrahedra containing the vertex V;
- $\mathcal{S}(E) \stackrel{\text{def}}{=} \{K \in \mathcal{T} : E \in K\}$ denotes the set of tetrahedra containing the edge E;
- $\mathcal{F}(F) \stackrel{\text{def}}{=} \{K \in \mathcal{T} : F \in K\}$ denotes the set of tetrahedra containing the face F.

The proposed intersection algorithm is described as pseudocode in Algorithm 2 and its main steps are the following:

I.1 **Localization.** We begin by localizing each solid mesh vertex $V \in \mathcal{T}_h^{\mathrm{s}}$ inside the fluid mesh $\mathcal{T}_h^{\mathrm{f}}$. The localization is carried out using a barycentric coordinates based algorithm, which efficiently identifies the element of the fluid mesh containing a given point (see, e.g., Frey and George (2007); Alauzet and Mehrenberger (2010)). This yields the map

$$\mathcal{M} : \mathcal{V}_{\mathcal{T}_h^{\mathrm{s}}} \to \mathcal{K}_{\mathcal{T}_h^{\mathrm{f}}} \cup \{\emptyset\} \qquad \mathcal{M}(V) \stackrel{\text{def}}{=} \begin{cases} K & \text{if } V \cap \mathcal{T}_h^{\mathrm{f}} \neq \emptyset, \\ \emptyset & \text{if } V \cap \mathcal{T}_h^{\mathrm{f}} = \emptyset, \end{cases}$$

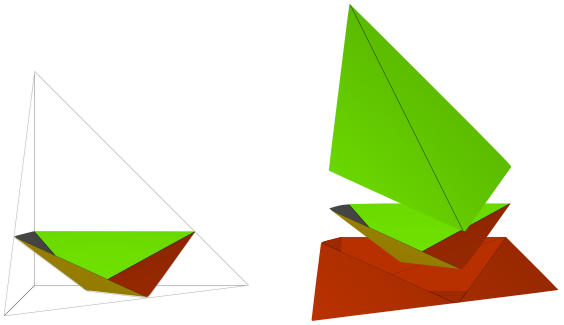
for all vertex $V \in \mathcal{V}_{\mathcal{T}_h^{\mathrm{s}}}$;

I.2 **Initialization.** We build the list of intersected fluid elements $\mathcal{L} = \{K \in \mathcal{T}_h^{\mathrm{f}} : \exists V \in \mathcal{V}_{\mathcal{T}_h^{\mathrm{s}}} \quad \text{s.t.} \quad \mathcal{M}(V) = K\}$. Moreover, for every $K \in \mathcal{L}$ we assemble $\mathcal{L}_K$ the list of solid triangles intersecting the fluid tetrahedron K, namely $\mathcal{L}_K = \{T \in \mathcal{T}_h^{\mathrm{s}} : \exists V \in T \quad \text{s.t.} \quad \mathcal{M}(V) = K\}$. These lists will be updated dynamically during the next steps of Algorithm 2. Moreover, for every $K \in \mathcal{L}$ we have: a local intersected fluid mesh $\mathcal{T}_K^{\mathrm{f}}$ initialized with the element K, a local intersected solid mesh $\mathcal{T}_K^{\mathrm{s}}$ (empty) and a local tolerance ε_K;

I.3 **Intersection.** We continue by looping over the fluid elements contained in $\mathcal{L}$. For each tetrahedron $K \in \mathcal{L}$, we intersect, one by one, the solid triangles contained in the list $\mathcal{L}_K$. For each triangle $T \in \mathcal{L}_K$, we begin by inserting in order its vertices, edges and face (see, respectively, Sub-steps I.3.1), I.3.2 and I.3.3 below). If an invalid configuration appears during this stage, the local tolerance ε_K is adjusted, the intersected meshes are reinitialized as in Step I.2, and the intersection process resumes from the first element in $\mathcal{L}_K$. Once the insertion stage has been completed, further checks are needed. For every solid edge E of T intersecting K, we add to $\mathcal{L}_K$ the element $\tilde{T} \in \mathcal{T}_h^{\mathrm{s}}$ sharing E with T (unless $\tilde{T}$ already belongs to $\mathcal{L}_K$). If the solid element T intersects the boundary face F of K, then, if not yet included, we add T to the list $\mathcal{L}_{\tilde{K}}$, where $\tilde{K}$ denotes the neighbour of K with respect to F in the mesh $\mathcal{T}_h^{\mathrm{f}}$. If $\tilde{K}$ has not already been included in $\mathcal{L}$, we add it. At this stage, we finish the intersection between K and T and we pass to the next solid element in $\mathcal{L}_K$. We point out that, since during the update some elements may have been added to the list $\mathcal{L}_{\tilde{K}}$ for some $\tilde{K}$ already processed, additional loops on the list $\mathcal{L}$ may be required;

I.4 **Consistency checks.** Once all the fluid elements have been intersected we check the consistency of the resulting intersected meshes with respect to some degenerate cases. Let assume, for instance, that in the fluid intersected mesh $\mathcal{T}_K^{\mathrm{f}}$ of $K \in \mathcal{T}_h^{\mathrm{f}}$, the element $T \in \mathcal{T}_h^{\mathrm{s}}$ is coplanar to the face F of K. Consequently, T must also be coplanar in the intersected mesh $\mathcal{T}_{\tilde{K}}^{\mathrm{f}}$, where $\tilde{K} \in \mathcal{F}(F) \setminus \{K\}$. Differently, two different local tolerances ε_K and $\varepsilon_{\tilde{K}}$ were used for $\mathcal{T}_K^{\mathrm{f}}$ and $\mathcal{T}_{\tilde{K}}^{\mathrm{f}}$, respectively. In this case, we keep between $\mathcal{T}_K^{\mathrm{f}}$ and $\mathcal{T}_{\tilde{K}}^{\mathrm{f}}$ the one obtained with the most restrictive tolerance, say $\mathcal{T}_K^{\mathrm{f}}$, so that we set $\varepsilon_{\tilde{K}} = \varepsilon_K$ and we intersect $\mathcal{T}_{\tilde{K}}^{\mathrm{f}}$ again. Let consider now the case of a vertex $V \in \mathcal{T}_h^{\mathrm{f}}$ whose counterpart in the intersected mesh $\hat{V} \in \mathcal{T}_K^{\mathrm{f}}$ lies exactly on an element $T \in \mathcal{T}_h^{\mathrm{s}}$, then we check whether this condition is verified for each $\mathcal{T}_{\tilde{K}}^{\mathrm{f}}$ with $\tilde{K} \in \mathcal{B}(V)$;

I.5 **Post-processing.** Finally, all the local intersected meshes $\mathcal{T}_K^{\mathrm{f}}$ (resp. $\mathcal{T}_K^{\mathrm{s}}$) are merged into a global one $\mathcal{T}_{\mathrm{Int}}^{\mathrm{f}}$ (resp. $\mathcal{T}_{\mathrm{Int}}^{\mathrm{s}}$). Additionally, the data structures needed for the element duplication and integration are updated accordingly. Regarding the latter, since for consistency only integration on the physical domain is considered, we need to define the notion of location of a sub-tetrahedron with respect to the interface. The solid separates the fluid mesh into two regions, which we will be referred to as *LEFT* and *RIGHT*. *LEFT* corresponds to the region and its sub-elements, in which the structure normal is pointing, while *RIGHT* is the opposite. This labelling starts from the sub-elements directly in contact with the solid, the remaining sub-elements are successively marked by proceeding neighbour by neighbour. Figure 9 provides an example of result of the algorithm.



(a) Initial configuration.

(b) Intersected solid mesh with sub-elements.

Figure 9: Example of intersected meshes and side coloured fluid intersected meshes. *LEFT* side: sub-elements coloured in green; *RIGHT* side: sub-elements in red.

We now provide a detailed description of the insertion steps mentioned in I.3. From this point forward, we will only be focussing on one fluid element in $\mathcal{T}_h^{\mathrm{f}}$, and we will be referencing to the selected tetrahedra in $\mathcal{L}$ with K, and the selected triangles in $\mathcal{L}_K$ with T. It should be noted that basic insertion operations, rather than complex vertex insertion operators (such as Delaunay kernel Frey and George (2000)), are carried out for efficiency. An example of the initial configuration with K and T is showed in Figure 10.

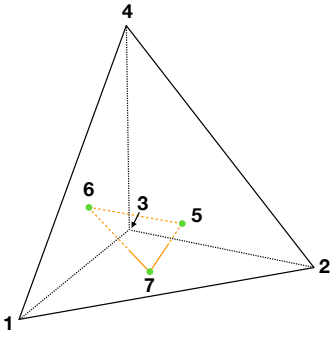


Figure 10: Initial configuration: fluid tetrahedron of vertices $1, 2, 3, 4$ and solid triangle $5, 6, 7$ partially intersecting the tetrahedron.

The core steps of the intersection algorithm are the following:

I.3.1 **Vertex insertion.** Let $V_i \in T$, with $i = 1, 2, 3$, be one of the three vertices of the solid triangle T. If V_i has already been treated while inserting the vertices of $\tilde{T} \in \mathcal{L}_K$, we move to the next vertex. Otherwise, V_i is inserted into the intersected meshes $\mathcal{T}_K^{\mathrm{f}}$ and $\mathcal{T}_K^{\mathrm{s}}$. Let $\hat{V}_i \in \mathcal{T}_K^{\mathrm{f}}$ be the copy of V_i in the intersected fluid mesh. Then, using the barycentric coordinates, we localize $\hat{V}_i$ with respect to K. If the vertex is outside the tetrahedron, i.e. $\hat{V}_i \cap K = \emptyset$, we start I.3.1 again for the next vertex of T. Otherwise, since in general the element K could

have been already intersected by another solid triangle, the vertex $\hat{V}_i$ is localized inside the intersected fluid mesh $\mathcal{T}_K^{\mathrm{f}}$ using the barycentric coordinates based algorithm adopted in step I.1. Once the vertex $\hat{V}_i$ is localized in an intersected fluid element $\hat{K} \in \mathcal{T}_K^{\mathrm{f}}$, four cases may arise (three of them being degenerated). The point may hence fall into:

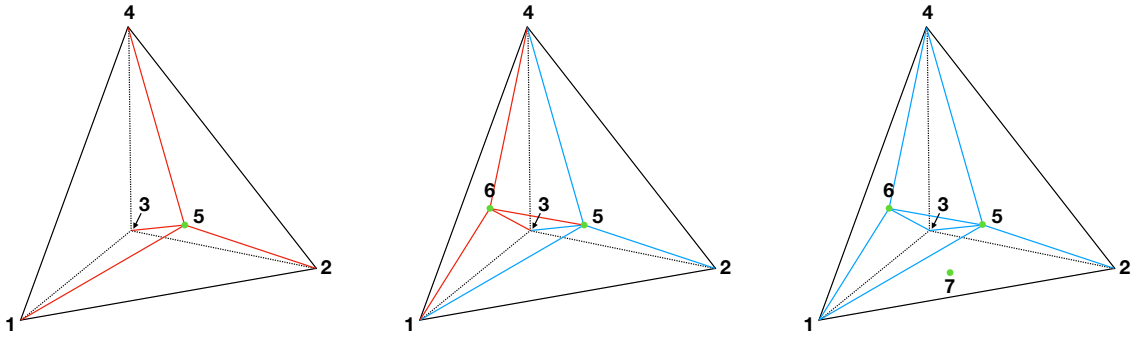
a) a vertex $\tilde{V}$ of $\hat{K}$, then $\hat{V}$ is replaced in every $\tilde{K} \in \mathcal{B}(\tilde{V})$;

b) an edge $\hat{E}$ of $\hat{K}$, then every $\tilde{K} \in \mathcal{S}(\hat{E})$ is divided in two sub-tetrahedra;

c) a face $\hat{F}$of $\hat{K}$, then every $\tilde{K} \in \mathcal{F}(\hat{F})$ is divided in three sub-tetrahedra;

d) the interior of $\hat{K}$, hence $\hat{K}$ is divided into four sub-tetrahedra.

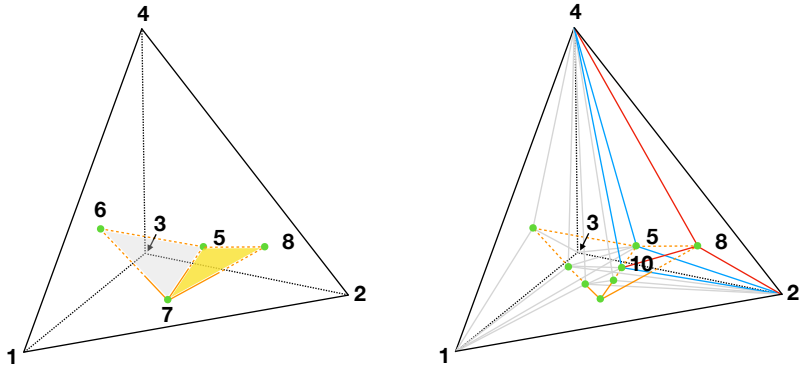
These steps are illustrated in Figure 11, starting from the configuration provided in Figure 10. For a more



(a) Internally. (b) On a face. (c) Outside.

Figure 11: Insertion of solid vertices inside the tetrahedron (I.3.1).

comprehensive illustration, Figure 12 shows a case in which K has already been intersected previously;



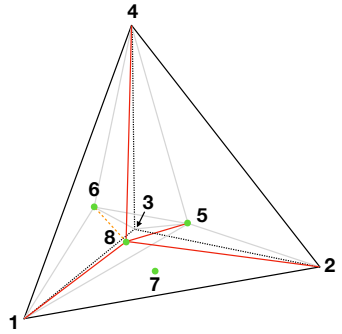
(a) Initial configuration.

(b) Insertion of vertex 8 inside sub-tetrahedron of vertices 2, 5, 10, 4.

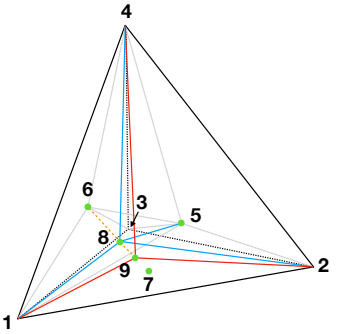
Figure 12: *Step* I.3.1 for solid triangle 8, 5, 7: insertion of solid vertices 8,5,7 inside tetrahedron $\{1, 2, 3, 4\}$, which is already been intersected from the solid triangle $\{5, 6, 7\}$. Vertices 5 and 7 are already been inserted, vertex 8 is internal to the sub-tetrahedron $\{2, 5, 10, 4\}$, hence, is divided in 4 sub-tetrahedra.

I.3.2 **Edges insertion.** Let $E \in T$ be one of the three edges of the solid triangle T, we proceed by inserting E into the intersected mesh $\mathcal{T}_K^{\mathrm{f}}$, by considering a partitioning algorithm (see, e.g., George et al. (2003) for a 2D version). Each edge E of T is treated one at a time. In case it has already been treated while inserting the edges of $\tilde{T} \in \mathcal{L}_K$, where $\tilde{T}$ is a solid triangle sharing E with T, we move to the next edge. Let V_1 and V_2 denote the two endpoints of E. Thanks to step I.3.1, we have that there exist $\hat{V}_1, \hat{V}_2 \in \mathcal{T}_K^{\mathrm{f}}$ such that $V_1 = \hat{V}_1$ and $V_2 = \hat{V}_2$.

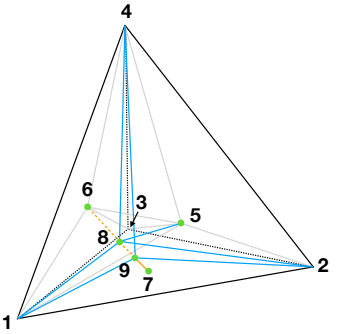
Starting from one of the two endpoints, for instance $\hat{V}_1$, we seek for the first face $\hat{F} \in \hat{K}$ such that $\hat{K} \in \mathcal{B}(\hat{V}_1)$ and $\hat{V}_1 \notin \hat{F}$, which is intersected by the edge $\hat{V}_1\hat{V}_2$, i.e., we search in all tetrahedra belonging to $\mathcal{B}(\hat{V}_1)$ whether there exists a face $\hat{F}$, opposite to vertex $\hat{V}_1$, that is intersected by the edge $\hat{V}_1\hat{V}_2$. The intersection point $\hat{V}_4$ is computed (see (Alauzet and Mehrenberger, 2010, Section 5), for details on the two-dimensional version) and inserted into the intersected meshes $\mathcal{T}_K^{\mathrm{f}}$ and $\mathcal{T}_K^{\mathrm{s}}$. Point $\hat{V}_4$ is necessarily on a tetrahedron face (internally to the face or in one of its edges/vertices) and therefore we apply the same insertion strategy of step I.3.1. At this stage the segment $\hat{V}_1\hat{V}_4$ has been added to the intersected mesh $\mathcal{T}_K^{\mathrm{f}}$. Then, the process is pursued by seeking for an intersection between the sub-edge $\hat{V}_4\hat{V}_2$ and a face $\hat{F} \in \hat{K}$ such that $\hat{K} \in \mathcal{B}(\hat{V}_4)$ and $\hat{V}_4 \notin \hat{F}$, which will give a new intersection point $\hat{V}_5$ and so on. If after n intersection steps the sub-edge $\hat{V}_{n+3}\hat{V}_2$ belongs to the intersected mesh $\mathcal{T}_K^{\mathrm{f}}$, the whole solid edge E has been inserted into $\mathcal{T}_K^{\mathrm{f}}$. Otherwise, the sub-edge $\hat{V}_{n+3}\hat{V}_2$ lies outside the fluid element K, and the intersection process stops. It is worth noting that to start the insertion process described above, at least one between $\mathcal{B}(\hat{V}_1)$ and $\mathcal{B}(\hat{V}_2)$ must be well defined, i.e., there exists $\hat{K} \in \mathcal{T}_K^{\mathrm{f}}$ such that $\hat{V}_1 \in \hat{K}$ or $\hat{V}_2 \in \hat{K}$. If the previous condition is not satisfied, we look for an intersection point $\hat{V}_4$ between the edge $\hat{V}_1\hat{V}_2$ and the faces of the tetrahedron K. The point $\hat{V}_4$, if it exists, is localized on the boundary sub-faces of the intersected fluid mesh $\mathcal{T}_K^{\mathrm{f}}$ using a barycentric coordinates based algorithm and finally inserted in $\mathcal{T}_K^{\mathrm{f}}$, applying the same strategy of step I.3.1, and in $\mathcal{T}_K^{\mathrm{s}}$. See Figure 13 for a detailed representation starting from the configuration of Figure 11;



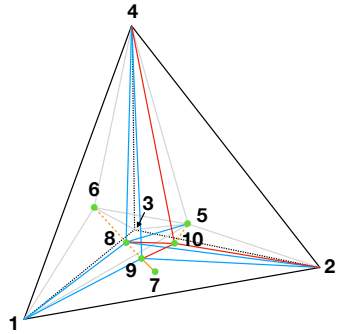
(a) Edge 6,7 intersects face 1,5,4 generating point 8.



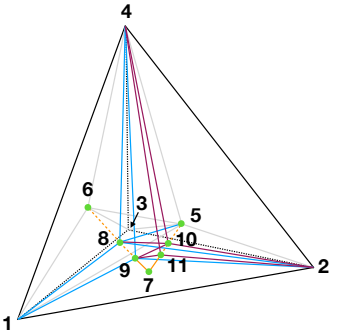
(b) Edge 8,7 intersects face 1,2,4 generating point 9.



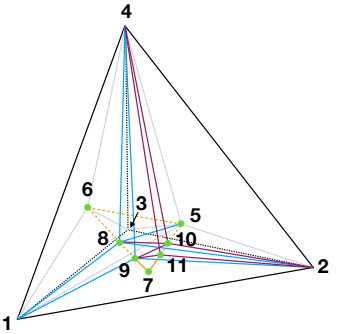
(c) Edge 9,7 is outside tetrahedron 1,2,3,4.



(d) Edge 5,7 intersects face 8,2,4 generating point 10.



(e) Edge 10,7 intersects face 9,2,4 generating point 11. Edge 11,7 is outside tetrahedron 1,2,3,4.

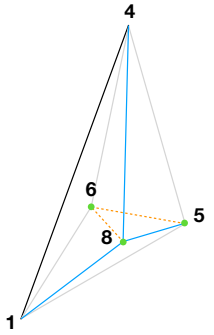


(f) Edge 5,6 is already included in the intersected mesh.

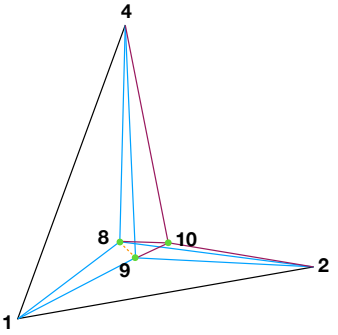
Figure 13: Insertion of the structure edges into the local intersected mesh resulting from step I.3.1. Insertion of edges $6,7$ (a)-(c), edge $5,7$ (d)-(e) and edge $5,6$ (f).

I.3.3 **Face insertion.** At this stage, we reconstruct the portion of the solid triangle T (as union of its sub-elements) within the fluid intersected mesh. Thus, we search the solid sub-triangles among the fluid sub-faces. Let denote by $\mathcal{U} = \{\hat{V}_i\}_{i=1}^n$ the union of the three vertices of T inserted in $\mathcal{T}_K^{\mathrm{f}}$ (i.e., $\hat{V}_i$ with $i = 1, 2, 3$) with all the vertices

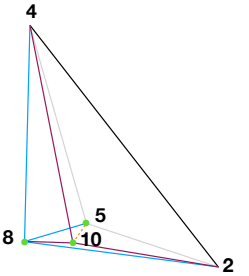
generated intersecting the edges of T in step I.3.2 (i.e., $\hat{V}_i$ with $i = 4, \ldots, n$). Moreover we denote by $\mathcal{E}$ the set of sub-edges generated while intersecting the edges of T in step I.3.2. Beginning from a sub-edge $\hat{E}$ with endpoints $\hat{V}_i$ and $\hat{V}_{i+1}$ and proceeding for all sub-edges belonging to $\mathcal{E}$, we search between all the sub-tetrahedra $\hat{K} \in \mathcal{S}(\hat{E})$ one with a face of vertices $\hat{V}_i\hat{V}_{i+1}Q$, with $Q \in \mathcal{U}$. In case a sub-tetrahedron with aimed vertices is detected, we insert the triangle $\hat{V}_i\hat{V}_{i+1}Q$ in $\mathcal{T}_K^{\mathrm{s}}$. We report in Figure 14 these steps starting from the configuration of Figure 13. On the contrary, if the sub-tetrahedron with aimed vertices is not detected, we still have to recover



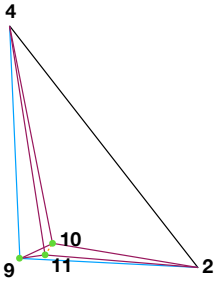
(a) Searching between the shell of edge $6, 8$ we select triangle $6, 8, 5$.



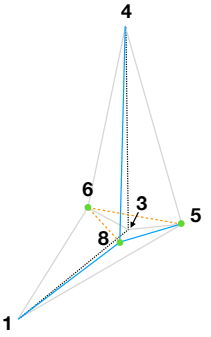
(b) Searching between the shell of edge $8, 9$ we select triangle $8, 9, 10$.



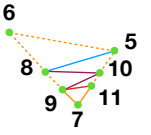
(c) Searching between the shell of edge $5, 10$ we select triangle $5, 10, 8$.



(d) Searching between the shell of edge $10, 11$ we select triangle $10, 11, 9$.



(e) Searching between the shell of edge $5, 6$ we select triangle $5, 6, 8$.



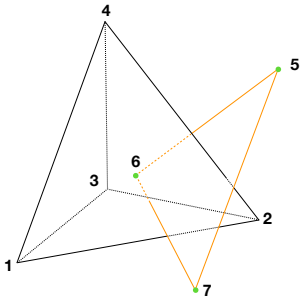
(f) Recovered intersected structure triangle $5, 6, 7$.

Figure 14: Insertion of structure face $5, 6, 7$ and recovery of the interface from the intersected sub-faces.

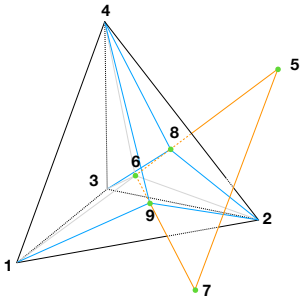
the surface. Therefore, we look for an edge $\tilde{E} \in \hat{K}$ such that $\hat{K} \in \mathcal{S}(\hat{E})$ and $\tilde{E} \cap \hat{E} = \emptyset$ which intersects the solid triangle T, i.e., we search in every tetrahedron belonging to $\mathcal{S}(\hat{E})$ whether there exists an edge $\tilde{E}$ opposite to $\hat{E}$ intersecting T. If $\tilde{E}$ exists, we insert the intersection point Q as in step I.3.1 in $\mathcal{T}_K^{\mathrm{f}}$ and $\mathcal{T}_K^{\mathrm{s}}$. Figure 15 provides an example of this situation. In order to correctly perform the labelling step in I.5, it is crucial that each new triangle $\hat{V}_i\hat{V}_{i+1}Q$ and T have the same orientation. In both cases, sub-edges $\hat{V}_{i+1}Q$ and $Q\hat{V}_i$ are included into $\mathcal{E}$, if a sub-edge is included twice then it is removed from $\mathcal{E}$ since both sub-triangles sharing the sub-edge have been recovered. When $\hat{E} = \hat{V}_{i+1}Q$ or $\hat{E} = Q\hat{V}_i$ is treated again the tetrahedra having the face with vertices $\hat{V}_i\hat{V}_{i+1}Q$ are removed from $\mathcal{S}(\hat{E})$. It is worth noting that $\mathcal{E} = \emptyset$ does not imply $T \cap K = \emptyset$ (e.g. the case where T intersects K while having all its edges outside of it). Consequently, we check for intersections between T and the edges of K. If no intersection is detected then the triangle T does not intersect the tetrahedron K. Otherwise each intersection point P_i (with $1 \leq i \leq 4$) is added into $\mathcal{T}_K^{\mathrm{f}}$ and $\mathcal{T}_K^{\mathrm{s}}$ following the procedure in I.3.1 and four cases may arise according to the number N of intersection points detected:

a) $N = 1$, the intersection process for T stops here;

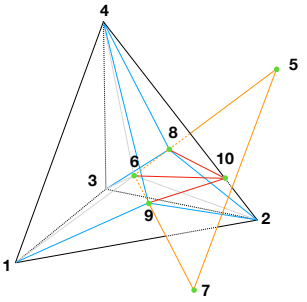
b) $N = 2$, the intersection process for T ends by performing step I.3.2 on the egde P_1P_2;



(a) Initial configuration.



(b) Vertices and edges insertion.



(c) Interface reconstruction. Intersection between edge $2, 4$ and triangle $6, 7, 5$ generates vertex 10, which is inserted. Sub-triangles $6, 9, 10$ and $6, 10, 8$ are now available.

Figure 15: Reconstruction of structure sub-triangles after further solid triangle and fluid edges intersection.

c) $N = 3$, a new triangle $\tilde{T} = P_1P_2P_3$ is created by preserving the orientation of T and steps I.3.2 and I.3.3 are performed on $\tilde{T}$;

d) $N = 4$, step $N = 3$ is applied to the two triangles $\tilde{T}_1 = P_1P_2P_3$ and $\tilde{T}_2 = P_1P_3P_4$.

Iterating over all the sub-edges contained in $\mathcal{E}$, we recover all the surface portion of T intersected by K;

In the case of partially intersected fluid domain the algorithm remains unchanged, except for the following steps:

- In step I.1, the vertices belonging to the fictitious interface Σ^n_{fic} are not localized inside the fluid mesh $\mathcal{T}^{\text{f}}_h$. Therefore the fluid elements containing only vertices belonging to fictitious triangles are not included in the list $\mathcal{L}$;
- In step I.3 after the insertion of vertices (I.3.1), edges (I.3.2) and face (I.3.3), if T intersects the boundary face F of K, but it belongs to the fictitious interface Σ^n_{fic}, we do not add it in the list $\mathcal{L}_{\tilde{K}}$, where $\tilde{K}$ is the neighbour element of K with respect to F. Thus the use of fictitious interface elements for intersection purpose remains confined to the elements containing the physical interface.

4.2. Element duplication

The duplication procedure is carried out as follows. The overlapping meshes $\mathcal{T}^{\text{f},n}_{h,1}$ and $\mathcal{T}^{\text{f},n}_{h,2}$ are created starting from a conforming triangulation $\mathcal{T}^{\text{f}}_h$ of the whole fluid domain Ω and duplicating the elements that are intersected by the interface Σ^n. Using the information coming from Algorithm 2 described in Section 4.1, let

$$\mathcal{G}_h \overset{\text{def}}{=} \{K \in \mathcal{T}^{\text{f}}_h : K \cap \Sigma^n \neq \varnothing\}$$

be the subset of elements intersected by Σ^n. For each element $K \in \mathcal{G}_h$, with nodes $\{i, j, k, l\}$, we can identify two regions $\Omega^{\text{f},n}_1$ and $\Omega^{\text{f},n}_2$, separated by the interface Σ^n, see Figure 16. In what follows, we will distinguish between *physical* and *non-physical* domains. The former consists of the restriction of $\mathcal{T}^{\text{f}}_{h,i}$ to $\Omega^{\text{f},n}_i$, $i = 1, 2$, and the latter corresponds to the complementary part. Let consider a duplication $\{i', j', k', l'\}$ of nodes $\{i, j, k, l\}$. We define two identical copies of K, namely K_1 and K_2, such that they will identify one of these regions. These new elements K_1,K_2 contain two parts, namely *physical* (grey in Figure 16) and *non-physical*, and they are defined such that if an original node $s \in \Omega^{\text{f},n}_i \subseteq K$, $i = 1, 2$, then $s \in \mathcal{T}^{\text{f}}_{h,i}$. More precisely, the original nodes are kept on the physical part of the duplicated elements, i.e., they remain on each side of the interface. As a result, the duplicated nodes lie in the *non-physical* side of each K_i, $i = 1, 2$. Suppose that node l is in $\Omega^{\text{f},n}_1$ whereas i, j, k are in $\Omega^{\text{f},n}_2$ (see Figure 16). After the duplication process we have $l \in \mathcal{T}^{\text{f}}_{h,1}$ and its duplicated node $l' \in \mathcal{T}^{\text{f}}_{h,2}$, contrarily $i, j, k \in \Omega^{\text{f},n}_2$, we have $i, j, k \in \mathcal{T}^{\text{f}}_{h,2}$ and $i', j', k' \in \mathcal{T}^{\text{f}}_{h,1}$ (see Figure 16). In order to preserve the continuity between two physical adjacent duplicated elements, if a node has been already duplicated, we select the pre-existing duplication as duplicated node. To summarize, we build two

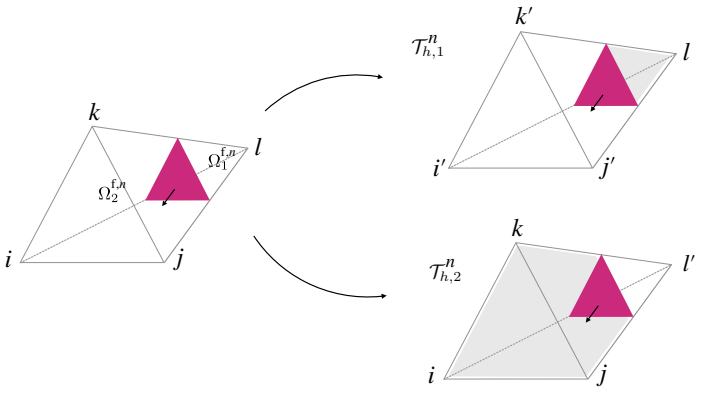


Figure 16: Duplication of tetrahedron $\{i, j, k, l\}$ (left) into $\{i, j, k, l'\}$ and $\{i', j', k', l\}$ (right).

copies of the intersected fluid elements, duplicating their nodes and keeping the original nodes on the physical side of each element. The result of this process is two independent meshes $\mathcal{T}_{h,1}^{\mathrm{f}}$ and $\mathcal{T}_{h,2}^{\mathrm{f}}$ covering $\Omega_1^{\mathrm{f},n}$ and $\Omega_2^{\mathrm{f},n}$, respectively, designed in such a way that the correct connectivity of the meshes is guaranteed.

In the case of partially intersected fluid domain, the duplication procedure remains unchanged. As a result, a fluid-fluid discontinuity is created and which is treated in a DG fashion (see Section 3.3). Suppose that tetrahedron $\{i, j, k, l\}$ is a tip element and its resulting duplicated elements are $\{i, j, k, l'\}$ and $\{i', j', k', l\}$ (see Figure 17). We first analyse

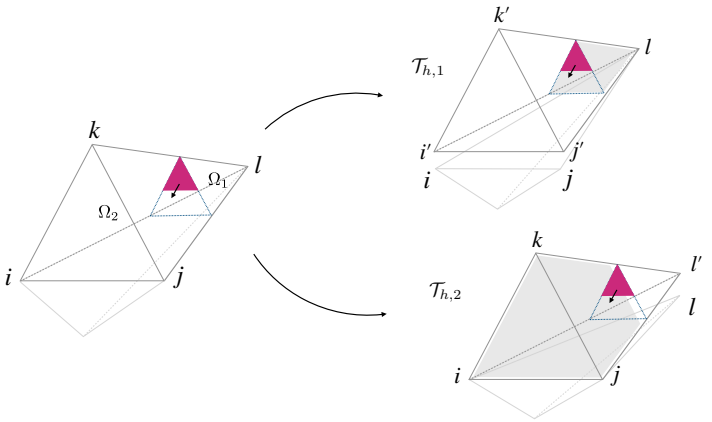


Figure 17: Duplication of tetrahedron $\{i, j, k, l\}$ containing a structure front (left). The resulting duplicated elements are $\{i, j, k, l'\}$ and $\{i', j', k', l\}$ (right). The below neighbour is included to show the loss of continuity on the bottom face.

the connection with the neighbouring elements. Faces whose opposite element is also duplicated are well connected with the adjacent element as a result of the duplication process, consider for instance faces $\{i, k, l'\}$ and $\{j', l', k'\}$ in Figure 17. Faces whose opposite element is not duplicated are completely disconnected or partially connected with the neighbour, (i.e., faces $\{i, l', j\}$ and $\{i', l, j'\}$ in Figure 17 are not connected with face $\{i, l, j\}$). Hence the need of a DG treatment between physical regions of disconnected faces (*tip faces*) in order to guarantee the o continuity between the two fluid sides.

5. Numerical experiments

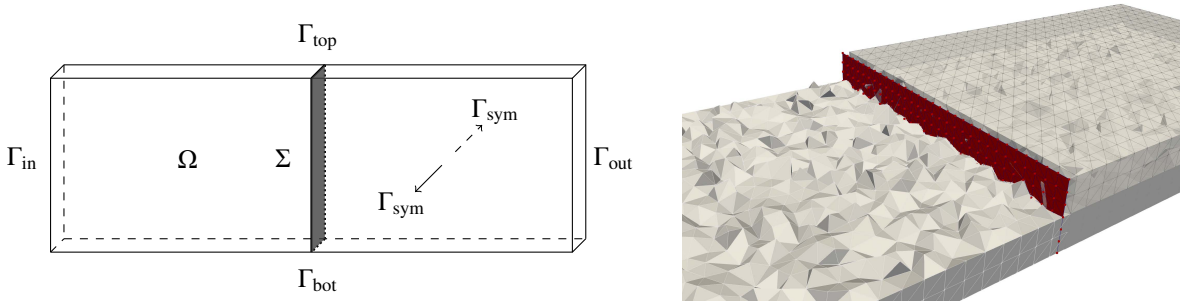
In this section, we provide a detailed numerical study with the purpose of illustrating the behavior of Algorithm 1. Four different examples involving dynamic interfaces and both partially and fully intersected fluid domains are discussed. The first two examples (§5.1 and §5.2) have a planar symmetry so that, for validation purposes, the obtained

results are compared with their corresponding 2D counterpart. The other two numerical examples are purely 3D problems. Along this section the user-defined parameters in Algorithm 1 are fixed to $\gamma_g = 1$, $\gamma_p = 10^{-2}$, $\gamma_v = 10^{-2}$, $\gamma = 100$. All the physical units are given in the CGS system.

A Reissner-Mindlin type shell model has been considered for the solid, with a spatial discretization based on linear triangular MITC3 elements with 5 degrees of freedom per node in the increments (see, e.g., Chapelle and Bathe (2011)). The coupled problem (10) involved in Algorithm 1 has been solved with a (parameter free) partitioned solution procedure, based on interface Newton-GMRES Dirichlet–Neumann iterations (see, .e.g, Fernández and Moubachir (2005); Deparis et al. (2006)). The last two numerical examples (§5.3 and §5.4) involve fully enclosed fluids. Therefore, a volumetric constraint is enforced on the solid velocity (in Algorithm 1) in order to guarantee its compatibility (Dirichlet data for the fluid) with the incompressibility of the fluid during the Dirichlet−Neumann iterations (see, e.g., Küttler et al. (2006)). All the computations have been performed with the `FELiScE` finite element library[1].

5.1. Idealized closed valve

The first example reproduces the behaviour of a closed valve under a pressure drop. This test case is a classical benchmark problem (see, e.g., Kamensky et al. (2015); Van Loon et al. (2004)) extensively tested with the two-dimensional version of Algorithm 1 in Boilevin-Kayl et al. (2019b); Alauzet et al. (2016). The problem consists of an



(a) Geometric configuration. (b) Fluid and solid meshes.

Figure 18: Geometric description and computational meshes.

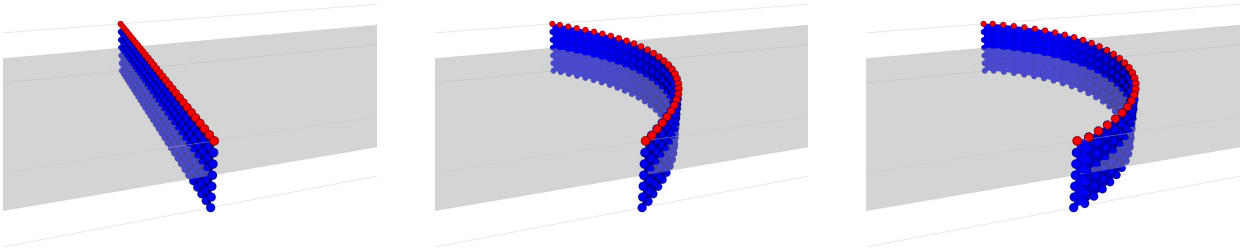
elastic shell clamped on its edges on Γ_{top} and Γ_{bot} and immersed in a channel filled with an incompressible Newtonian fluid, as shown in Figure 18(a). The fluid domain is given by $\Omega = [0, 3] \times [0, 1] \times [0, 0.2]$. The reference solid configuration is $\Sigma = \{1.5\} \times [0, 1] \times [0, 0.2]$. As regards the boundary conditions for the fluid, a no-slip boundary condition is enforce on Γ_{top} and Γ_{bot}. Zero traction is enforced on the outer boundary Γ_{out}, while on Γ_{in} the following time-dependent pressure function $p_{in}(t)$ is prescribed:

$$p_{in}(t) = -10^5 \cdot \tanh(10\, t), \quad t \in \mathbb{R}^+.$$

In addition, a symmetry condition, namely, $\boldsymbol{u} \cdot \boldsymbol{n} = 0$ and $\boldsymbol{\sigma}(\boldsymbol{u}, p)\boldsymbol{n} \cdot \boldsymbol{\tau} = \boldsymbol{0}$, is enforced on the lateral walls Γ_{sym}. The solid is fully clamped at its extremities $\Gamma_{top} \cup \Gamma_{bot}$ and a symmetry condition is enforced on its lateral boundary (lying in Γ_{sym}), viz., the normal displacement and the rotation are set to zero. Both the fluid and the solid are initially at rest. The physical parameters for the fluid are: density $\rho^f = 1.0$ and dynamic viscosity $\mu = 0.035$. For the solid, we have: density $\rho^s = 1.1$, Young's modulus $E = 7.5 \cdot 10^5$, thickness $\varepsilon = 0.1$ and Poisson's ratio $\nu = 0.5$. The fluid and solid meshes are shown in Figure 18(b), with a local size of $h \approx 0.05$. The fluid mesh is made of 47181 tetrahedra, while tor the solid we have 360 triangles. The time-step is $\tau = 10^{-3}$ and the final time is set to $T = 1$.

As the prescribed pressure increases, the solid starts to bend. After a brief dynamics phase, the system reaches a steady state with a constant pressure jump across the interface. The fluid and solid velocities vanish and the pressure is a piecewise constant. Some snapshots of the resulting deformed mesh at time $t = 0$, $t = 0.05$ and $t = 1$. are

[1] `https://gitlab.inria.fr/felisce/felisce`



(a) $t = 0$. (b) $t = 0.05$. (c) $t = 1$.

Figure 19: Deformed configurations in 2D (red dots) and 3D (blue) at different time instants.

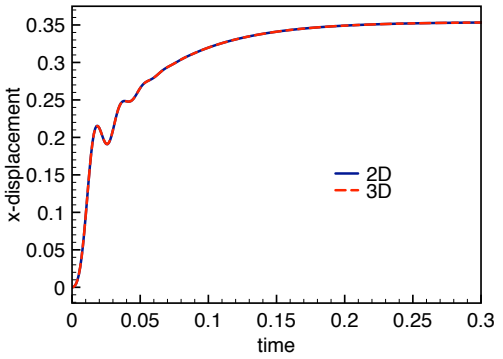
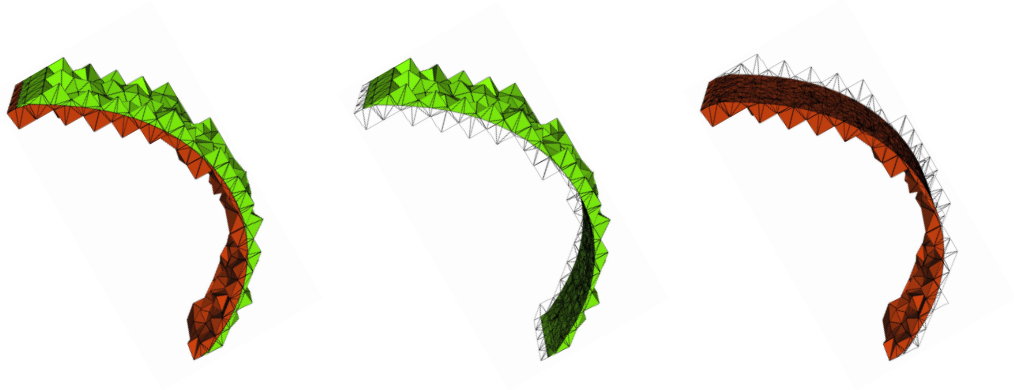


Figure 20: Solid x-displacement of the closed valve mid-point for the 2D-case and for the midpoint on the place $z = 0$ for the 3D.



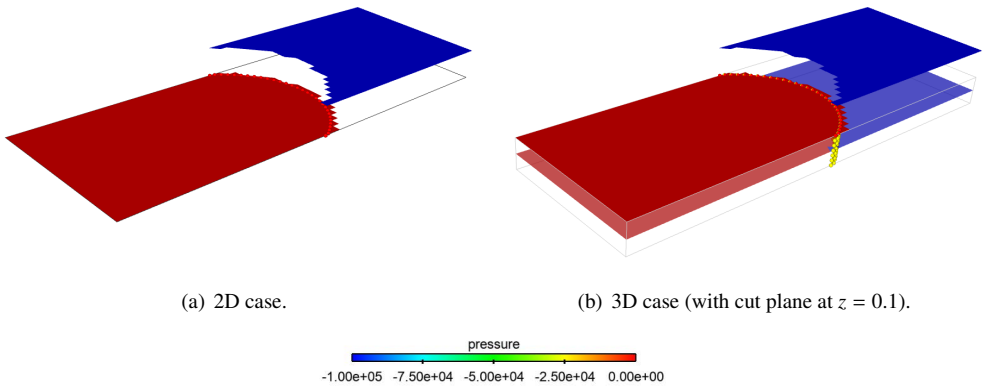
(a) Both side coloured. (b) *LEFT* side in green. (c) *RIGHT* side in red.

Figure 21: Fluid intersected mesh at $t = 1$.

shown in Figure 19. We can observe that the 2D deformed configuration (red dots) and the 3D (blue) perfectly match. Already at $t = 0.3$, the structure has reached the stationary states. Figure 20 reports a comparison of the time history of the solid x-displacement of the closed-valve mid-point. The two results are practically indistinguishable. For illustration purposes, Figure 21 and Figure 22 show the 3D intersected fluid and solid meshes respectively at the reached steady state. The fluid intersected mesh includes only the fluid elements that are cut by the deformed solid mesh (it contains 16896 tetrahedra). The two sides of the mesh, *LEFT* and *RIGHT*, are highlighted with different colours. As regards the solid, Figure 22(a) shows the deformed solid mesh, while Figure 22(b) shows the intersected mesh with its sub-elements (6805 triangles in this case).

(a) Deformed solid mesh. (b) Intersected deformed solid mesh.

Figure 22: Deformed and intersected solid meshes at $t = 1$.



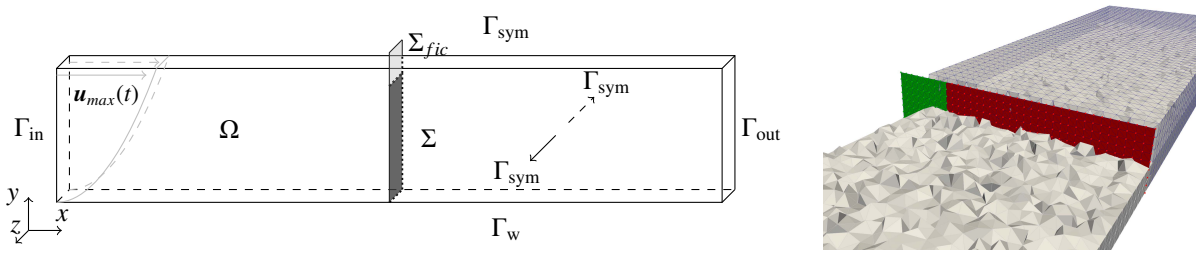
(a) 2D case. (b) 3D case (with cut plane at $z = 0.1$).

Figure 23: Snapshots of the pressure elevation at time $t = 1$.

For illustration purposes, pressure snapshots and elevations for the 2D and 3D solutions are reported in Figure 23. Both 2D and 3D Nitsche-XFEM methods get the correct pressure jump, 10^5 on one side and zero on the other.

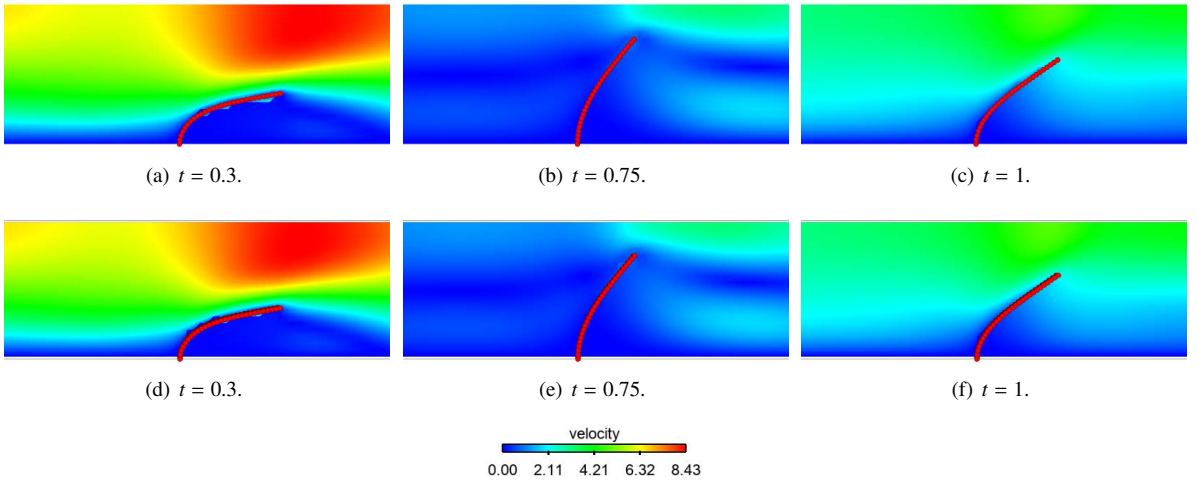
5.2. Idealized open valve

As a second example, we consider the heart-valve inspired benchmark problem proposed in Kamensky et al. (2015); Gil et al. (2013); Hesch et al. (2012); Wick (2013). In this case the solid has a boundary inside the fluid domain, so that the front treatment proposed in Section 3.3 is used. The fluid domain is given by $\Omega = [0, 4] \times$



(a) Geometric configuration. (b) Fluid and solid meshes.

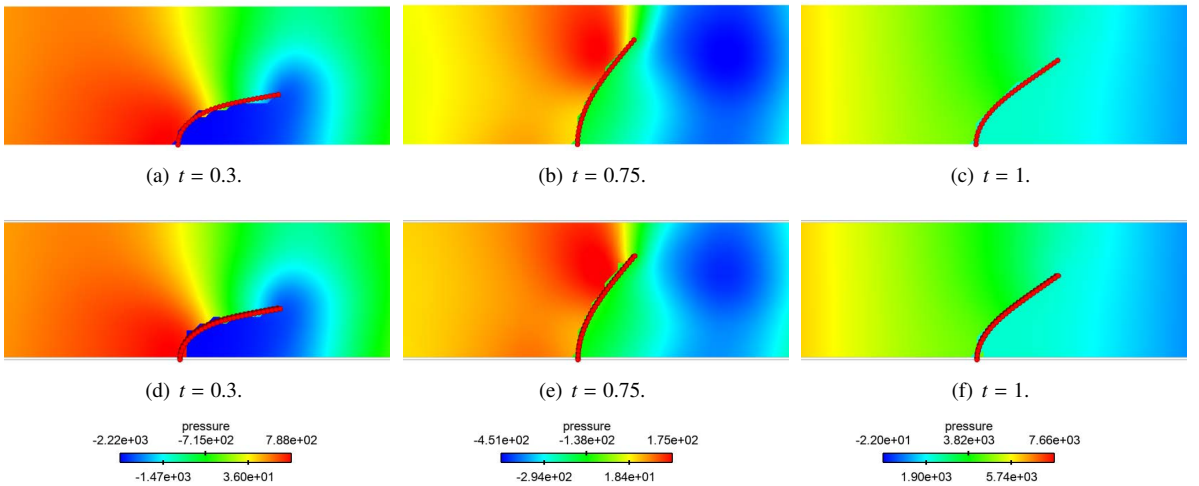
Figure 24: Geometrical description and computational meshes (the fictitious interface in green color).



(a) $t = 0.3$. (b) $t = 0.75$. (c) $t = 1$.

(d) $t = 0.3$. (e) $t = 0.75$. (f) $t = 1$.

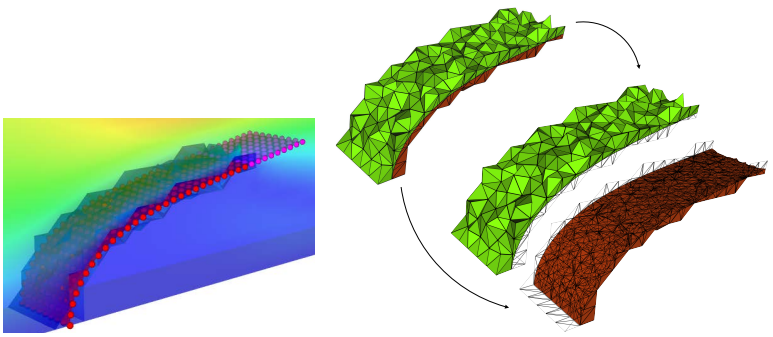
Figure 25: Snapshots of the fluid velocity magnitude at $t = 0.3$ (left column), $t = 0.75$ (central column) and $t = 1$. (right column) for both 2D (top row) and 3D (bottom row).



(a) $t = 0.3$. (b) $t = 0.75$. (c) $t = 1$.

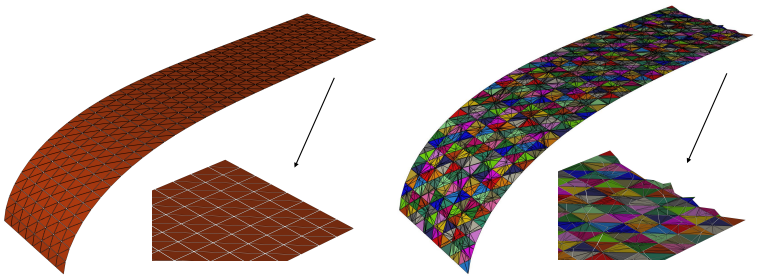
(d) $t = 0.3$. (e) $t = 0.75$. (f) $t = 1$.

Figure 26: Snapshots of the fluid pressure at $t = 0.3$ (left column), $t = 0.75$ (central column) and $t = 1$. (right column) for both 2D (top row) and 3D (bottom row).



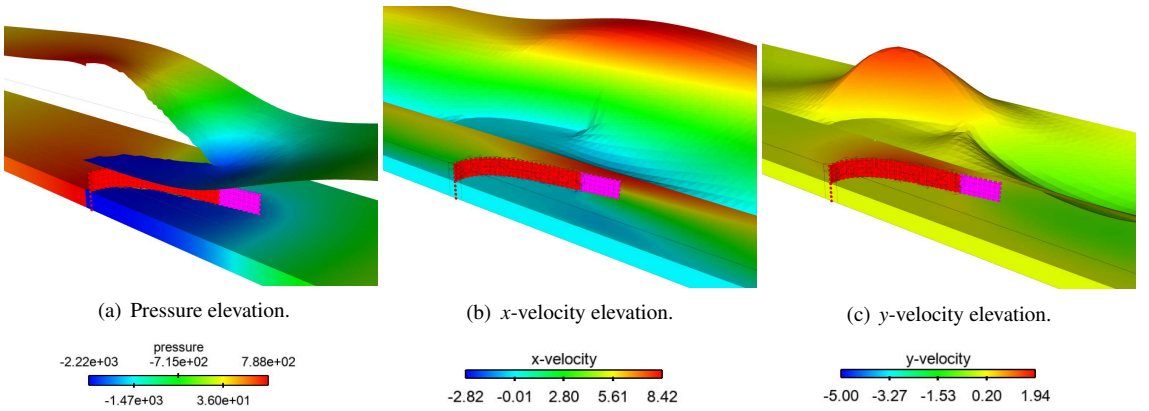
(a) Deformed solid mesh (Σ and Σ_{fic}). (b) Intersected fluid mesh.

Figure 27: Deformed solid interface (physical and fictitious) immersed in the fluid channel (a) and the fluid intersected mesh with its restriction on each side of the interface (b) at $t = 0.3$.



(a) Deformed solid mesh $\Sigma(t)$. (b) Intersected structure mesh

Figure 28: Deformed and intersected solid meshes at $t = 0.3$, both with a zoom on the immersed front solid boundary. Note that (a) contains only the physical interface $\Sigma(t)$ (the front is flat), while the front of (b) includes some sub-elements of Σ_{fic}.



(a) Pressure elevation. (b) x-velocity elevation. (c) y-velocity elevation.

Figure 29: Snapshots of the fluid elevated pressure, x-velocity and y-velocity of a clipped plane $z = 0.1$, obtained in the 3D case, at $t = 0.3$.

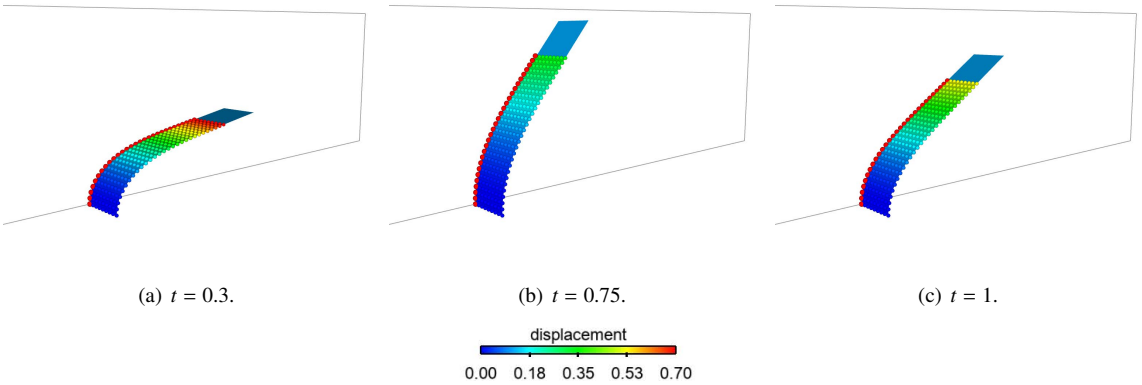
$[0, 0.805] \times [0, 0.2]$. For the structure we have $\Sigma = 2 \times [0, 0.7] \times [0, 0.2]$, and a fictitious interface $\Sigma_{\text{fic}} = 2 \times [0.7, 0.9] \times [0, 0.2]$. The considered geometry is shown in Figure 24(a). No-slip boundary condition is enforced on Γ_{w} and a symmetry condition is imposed on Γ_{sym} (top and lateral walls). Zero traction is imposed on Γ_{out} and a half parabolic profile is prescribed on Γ_{in}, with maximum velocity:

$$u_{\max}(t) = 5(0.805)^2(\sin(2\pi t) + 1.1), \quad t \in \mathbb{R}^+.$$

The solid is clamped at its bottom (i.e., displacement and rotations are set to zero). A symmetry condition is presribed on the lateral edges of the solid. Both fluid and solid are considered initially at rest. The physical parameters for the fluid in this test are: density $\rho^{\text{f}} = 100$ and dynamic viscosity $\mu = 10$. For the solid we have: density $\rho^{\text{s}} = 100$, thickness $\varepsilon = 0.0212$, Young's modulus $E = 5.6 \cdot 10^7$ and Poisson's ratio $\nu = 0.4$. The considered spatial discretization parameter is approximately $h \approx 0.04$. The fluid mesh is made of 102380 tetrahedra and the solid mesh of 576 triangles. Figure 24(b) shows the corresponding meshes. The physical region of the solid is coloured in red and the fictitious interface in green. The time-step is $\tau = 2 \times 10^{-3}$ and the final time is $T = 1$, which corresponds to one full oscillation cycle for the solid.

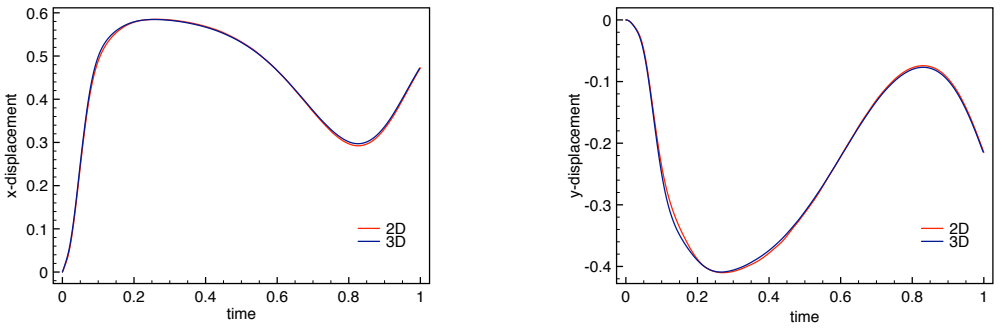
For illustration purposes, snapshots of the fluid velocity magnitude and the position of the interface, obtained in 2D and in 3D, are shown in Figure 25 at the time instants $t = 0.3$, 0.75 and 1, respectively. For the 2D case we show a zoom of the fluid domain near the deformed structure location, while for the three dimension case we plot the solution on the cut plane $z = 0.1$. A very good agreement is obtained for the two cases. Similarly conclusions can be inferred from Figure 26, which presents the snapshots of the pressure at the same time instants.

Figure 27(a) shows the deformed solid and the fictitious part at $t = 0.3$. Across the physical structure, we can see the faces of the intersected fluid mesh. Notice that no tetrahedra around the fictitious interface are visible, in fact they are not included into the fluid intersected mesh. In figure 27(b), we present the fluid intersected mesh, highlighting the two sides of the interface. The fluid intersected mesh is made of 17879 tetrahedra. Figure 28 reports the deformed



(a) $t = 0.3$. (b) $t = 0.75$. (c) $t = 1$.

Figure 30: Representation of the actual structure location at three different instants. The 2D solution (in red) is superimposed to the 3D (coloured by the solid displacement magnitude). The fictitious interface is displayed (coloured in blue).



(a) x-displacement. (b) y-displacement.

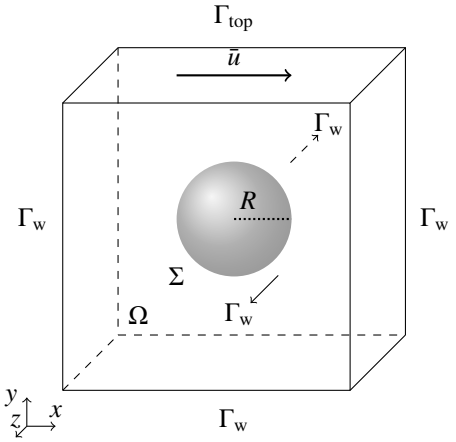
Figure 31: Displacement time history of the solid tip in 2D and the top node on the plane $z = 0$ in 3D.

solid mesh and the corresponding intersected surface mesh, which contains 7415 triangles. A zoom of the immersed front boundary is displayed for each mesh. The intersected structure mesh contains all the triangles of the original mesh (divided into sub-triangles) and the extra sub-triangles, belonging initially to Σ_{fic}, and which are necessary to close the the domain inside the fluid front elements. Figure 29(a) presents the pressure elevation for the cut plane $z = 0.1$. The fictitious part of the interface is included with the purpose of highlighting that the pressure discontinuity is well captured across the physical interface, while it is continuous across the fictitious interface, i.e., the fictitious interface is completely invisible for both the fluid and the solid. Figures 29(b) and (c) show respectively the elevation of the x- and y-component of the velocity. Each velocity component is continuous across Σ and also Σ_{fic}, but clearly no constraint on the velocity on Σ_{fic} is visible form the fluid side. The velocity z-component vanishes due to the symmetry of this problem.

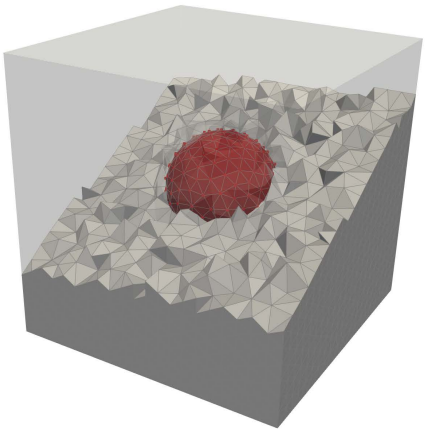
For illustration purposes, we show in Figure 30 the solid configurations in both the 2D and 3D cases with the displacement magnitude at the time instants $t = 0.3$, 0.75 and 1. Once more, no notable differences are visible between the 2D and 3D solutions. Figure 31 reports the x- and y-displacement time history of the upper 2D-solid endpoint and the upper 3D-solid on the plane $z = 0$. The 3D and 2D results are practically indistinguishable. Only very small differences are visible, which are expected to vanish after spatial refinement.

5.3. Spherical capsule in lid-driven cavity flow

In the following example, we consider the lid-driven cavity test with an immersed elastic spherical structure. The fluid domain is given by the unit cube $\Omega = [0, 1] \times [0, 1] \times [0, 1]$ and the reference solid configuration is a sphere of radius $R = 0.2$ centred at $x_c = 0.5$, $y_c = 0.5$, $z_c = 0.5$, as shown in Figure 32(a). A no-slip boundary condition is enforced on Γ_w and a constant velocity profile $\bar{\boldsymbol{u}} = \boldsymbol{e}_1$ is prescribed on Γ_{top}. Both the fluid and the solid are initially



(a) Geometrical configuration.

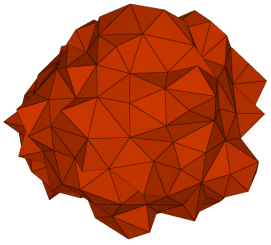


(b) Fluid and solid meshes.

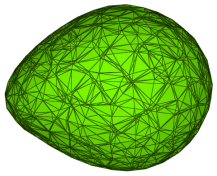
Figure 32: Geometric description and computational meshes.

at rest. The physical parameters for the fluid are: density $\rho^f = 100$ and dynamic viscosity $\mu = 10$. For the solid we have: density $\rho^s = 100$, thickness $\varepsilon = 0.0212$, Young's modulus $E = 5.6 \cdot 10^3$ and Poisson's ratio $\nu = 0.4$.

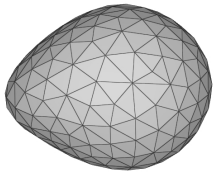
The time-step is set to $\tau = 5 \times 10^{-3}$ and the final time is $T = 10$. The fluid mesh is made of 28843 tetrahedra, while the solid mesh contains 446 triangles. Figure 32(b) shows the considered fluid and solid meshes. In both cases the space discretization parameter is $h \approx 0.05$. As discussed above, the intersection between the two meshes and the sub-tetrahedralization of the fluid intersected elements has to be evaluated at each time step. In particular, the solid mesh intersects on average 900 tetrahedra belonging to $\Omega^{f,n}$. The intersection Algorithm 2 produces a fluid mesh made of approximately 18000 tetrahedra and a solid mesh made of approximately 450 triangles. It should be noted that this induces additional computational cost (only) to the assembling phase, due to the integration over arbitrary polygons. Figures 33(a)–33(b) shows the 3D intersected fluid mesh at time $t = 5$ (17555 tetrahedra), where the two regions of the fluid domain are highlighted with different colours. Figure 33(c) shows the deformed solid mesh at $t = 5$ and Figure 33(d) the corresponding intersected solid mesh (6952 triangles). Each colour in Figure 33(d) indicates the original solid triangle.



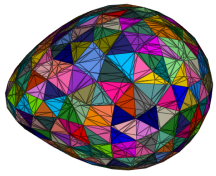
(a) Fluid intersected mesh (exterior side).



(b) Fluid intersected mesh (interior side).



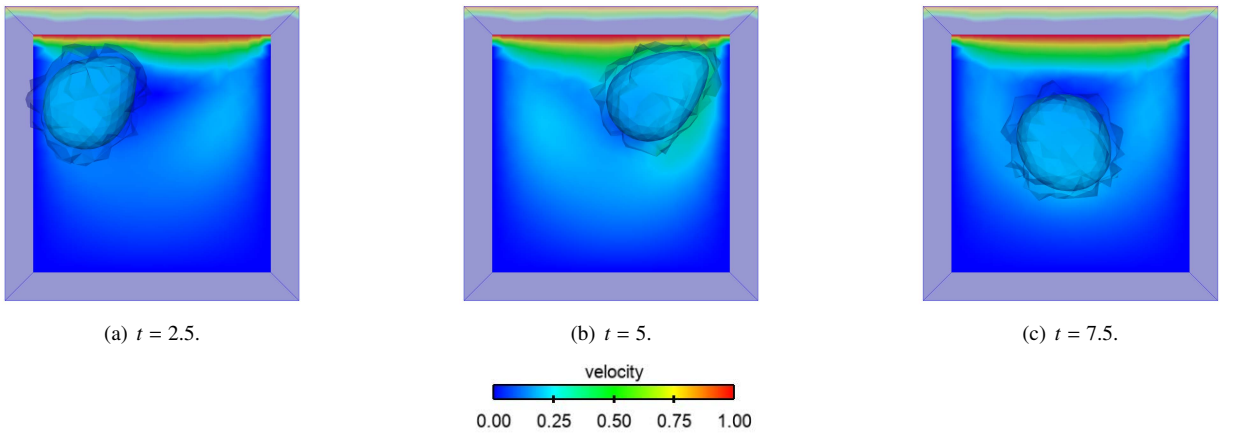
(c) Deformed solid mesh.



(d) Intersected deformed solid mesh.

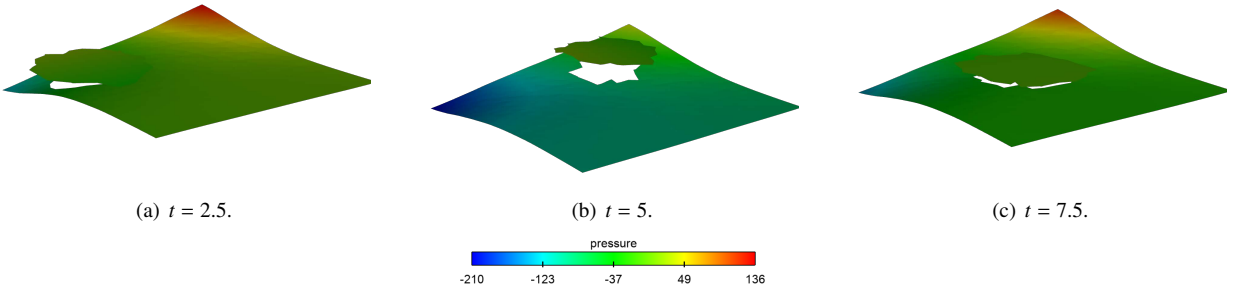
Figure 33: Fluid and deformed intersected solid meshes at $t = 2.5$.

For illustration purposes, Figure 34 shows the snapshots of the fluid velocity magnitude. At $t = 2.5$, Figure 34(a), the vesicle is heading towards the moving upper region of the cavity, getting closer to the end of the top boundary at $t = 5$, Figure 34(b). At $t = 7.5$, Figure 34(c), the vesicle has completed the first rotation around the initial position. Additionally, the solid current configuration is included for each time instant. Figure 35 shows the pressure elevation of the cutting plane $z = 0.5$ at the same time instants as for the velocity in Figure 34(a). The pressure jump across the moving interface Σ^n is clearly visible. Finally, in Figure 36 we present the trajectory projected on the xy plane, of the vesicle leftmost vertex, starting from $x = 0.3$ and $y = 0.5$, at $t = 0$. Time markers have been depicted in order to



(a) $t = 2.5$. (b) $t = 5$. (c) $t = 7.5$.

Figure 34: Velocity magnitude snapshots.



(a) $t = 2.5$. (b) $t = 5$. (c) $t = 7.5$.

Figure 35: Pressure snapshots and elevation.

facilitate the visualization of the result.

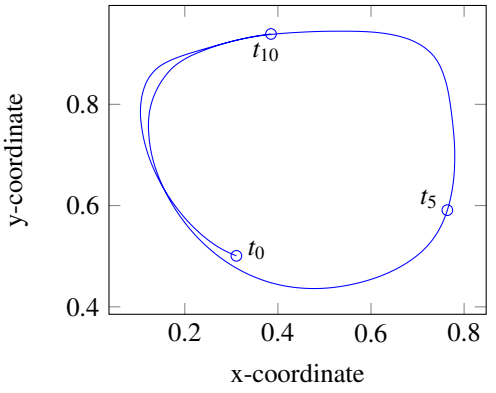
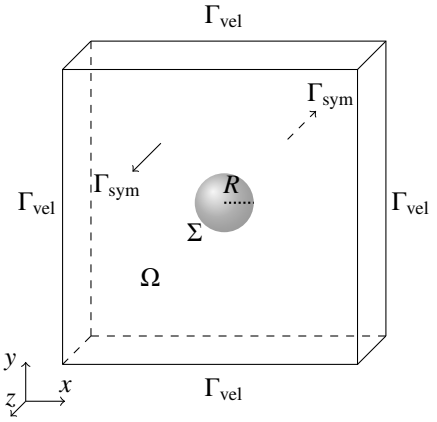


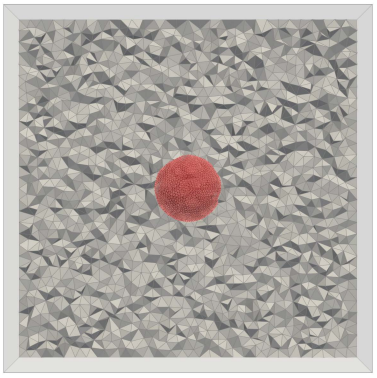
Figure 36: Trajectory of the leftmost solid node from $t = 0$ to $t = 10$.

5.4. Spherical capsule in shear flow

The last example is a classical benchmark problem in micro-encapsulation (see, e.g., Dupont et al. (2022)): the mechanical equilibrium of a liquid capsule enclosed by a thin-walled elastic solid immersed in a shear flow. The solid domain Σ is a spherical surface of radius $R = 0.3$, centered at the origin. The fluid domain is given by the prism $\Omega = [-5R, 5R] \times [-5R, 5R] \times [-2.5R, 2.5R]$, see Figure 37(a). On Γ_{vel} we impose the shear velocity profile

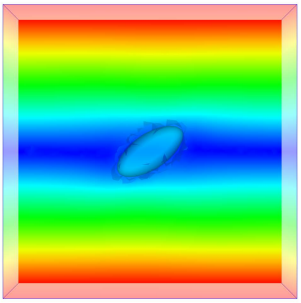


(a) Geometric description.

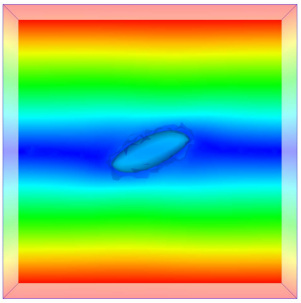


(b) Fluid and solid meshes.

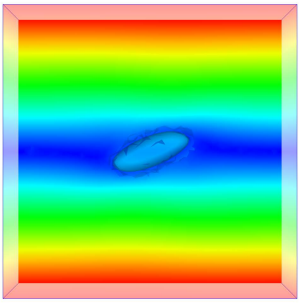
Figure 37: Geometric description and computational meshes.



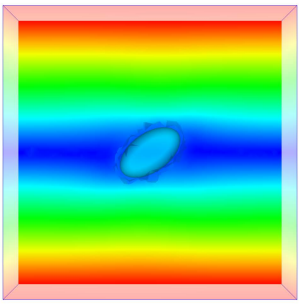
(a) $\mathrm{Ca_s} = 0.6$, $t = 1.0$.



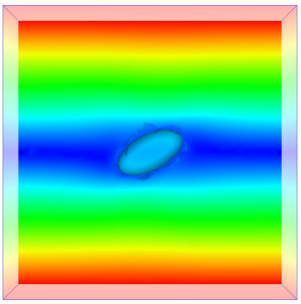
(b) $\mathrm{Ca_s} = 0.6$, $t = 3.0$.



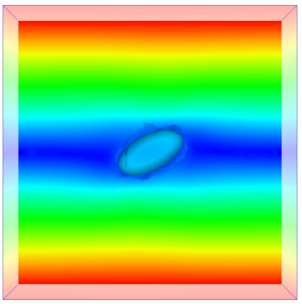
(c) $\mathrm{Ca_s} = 0.6$, $t = 6.0$.



(d) $\mathrm{Ca_s} = 0.18$, $t = 1.0$.



(e) $\mathrm{Ca_s} = 0.18$, $t = 3.0$.



(f) $\mathrm{Ca_s} = 0.18$, $t = 6.0$.

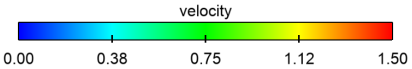


Figure 38: Velocity magnitude snapshots at three different time instants for $\mathrm{Ca_s} = 0.6$ (top) and $\mathrm{Ca_s} = 0.18$ (bottom).

$\boldsymbol{u}(x, y, z) = \dot{\gamma} y \boldsymbol{e}_1$, where the shear rate $\dot{\gamma}$ is set to 1. Symmetry conditions are imposed on the lateral walls Γ_{sym}. Both the fluid and the solid are initially at rest. The physical parameters for the fluid (inside and outside the capsule) are: density $\rho^{\mathrm{f}} = 1$ and dynamic viscosity $\mu = 1$. For the solid we take: density $\rho^{\mathrm{s}} = 1$, Young's modulus $E = 50$ and Poisson's ratio $\nu = 0.5$. A dimensional analysis shows that the two main quantities that drive the capsule dynamics are (see, e.g., Dupont et al. (2022)): the relative thickness $\alpha = \varepsilon/R$ and the bulk capillary number

$$\mathrm{Ca_v} \stackrel{\mathrm{def}}{=} \frac{\mu \dot{\gamma}}{G},$$

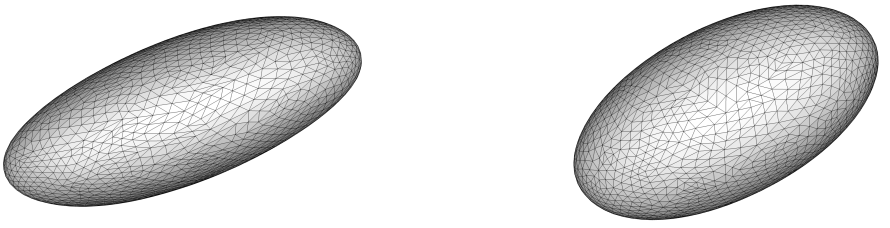
which represents the ratio between the viscous and the elastic forces. Here, the symbol $G \stackrel{\text{def}}{=} E/(2(1+\nu))$ denotes the solid shear modulus. It is also customary to introduce the surface capillary number

$$\mathrm{Ca}_\mathrm{s} \stackrel{\text{def}}{=} \frac{\mathrm{Ca}_\mathrm{v}}{\alpha}.$$

Owing to the physical parameters provided above, we have $\mathrm{Ca}_\mathrm{v} = 0.06$. In what follows, we will use Algorithm 1 to simulate the effect on the capsule dynamics of two different values of the solid thickness: $\varepsilon = 0.03$ and $\varepsilon = 0.1$, which respectively yield $\alpha = 0.1, \mathrm{Ca}_\mathrm{s} = 0.6$ and $\alpha = 0.\bar{3}, \mathrm{Ca}_\mathrm{s} = 0.18$.

The fluid and solid domain are discretized by means of two unstructured meshes with local size $h_\mathrm{f} \approx 0.05$ and $h_\mathrm{s} \approx 0.025$ respectively. The fluid mesh is made of 144130 tetrahedra and the solid mesh of 4944 triangles, see Figure 37(b). The time-step length is set to $\tau = 2 \cdot 10^{-3}$ and the final time to $T = 10$.

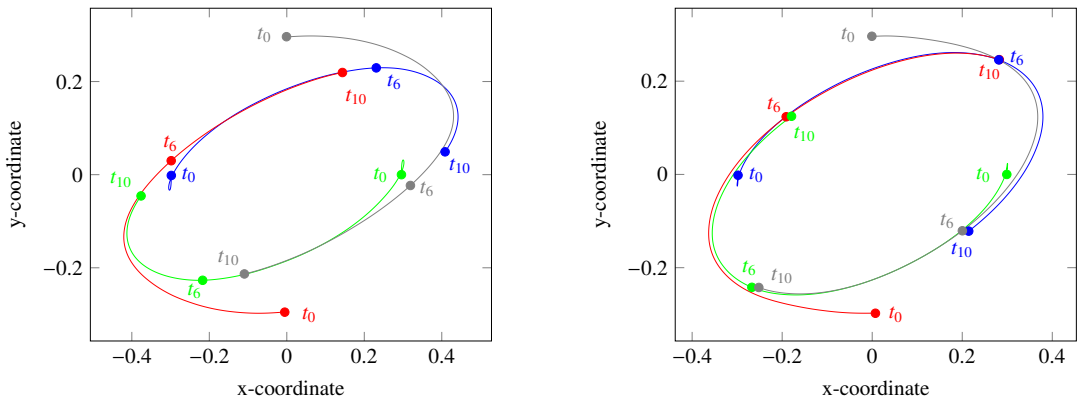
For illustration purposes, Figure 38 shows the snapshots of the fluid velocity magnitude at time $t = 1$, 3 and 6 on the cutting plane $z = 0$. In particular, the Figure 38 clearly shows the influence of the thickness on the capsule deformed shape. As expected, the thinnest capsule ($\mathrm{Ca}_\mathrm{s} = 0.6$) becomes much more elongated at equilibrium than the thickest one ($\mathrm{Ca}_\mathrm{s} = 0.18$). This can also be inferred from Figure 39, which shows the deformed configuration of the capsules at time $t = 10$. It is worth noting that, in the early stages of the simulation ($t = 1$, Figures 38(a) and (d))



(a) $\mathrm{Ca}_\mathrm{s} = 0.6$ (b) $\mathrm{Ca}_\mathrm{s} = 0.18$

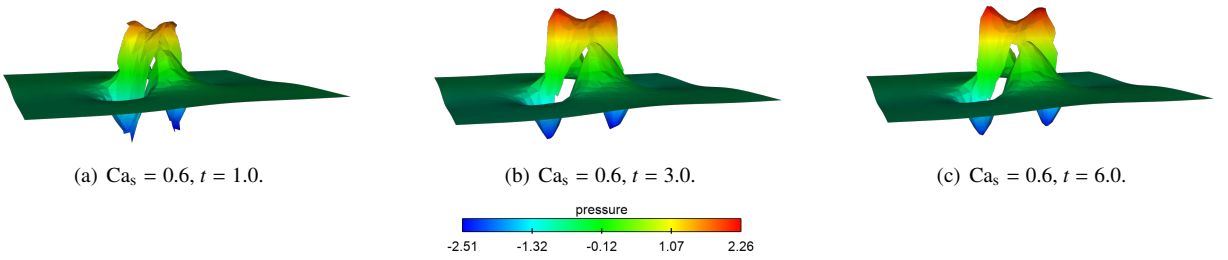
Figure 39: Deformed capsule at $t = 10$.

the capsule motion is mainly driven by the deformation due to the stresses exerted by the external flow. Progressively ($t = 3$, Figures 38(b) and (e)), this local deformation is transformed into a rotational motion along the deformed shape. Finally ($t = 6$, Figures 38(c) and (f)), the capsule reaches a steady state with a pure rotational motion. This behaviour can clearly be inferred from Figure 40 which shows the trajectories of four different points on the mid-plane $z = 0$. Finally, Figures 41 and 42 show the pressure elevation of the cutting plane $z = 0$ for both the surface capillary numbers $\mathrm{Ca}_\mathrm{s} = 0.6$ and $\mathrm{Ca}_\mathrm{s} = 0.18$, at time $t = 1$, 3 and 6. The pressure jump across the moving interface is well captured in both cases.



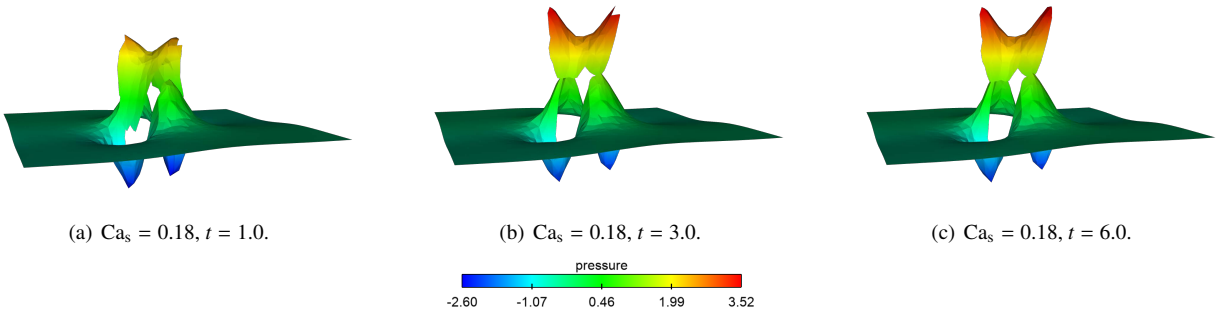
(a) $\mathrm{Ca}_\mathrm{s} = 0.6$. (b) $\mathrm{Ca}_\mathrm{s} = 0.18$.

Figure 40: Solid nodes trajectories projected on the plane $z = 0$.



(a) $Ca_s = 0.6$, $t = 1.0$. (b) $Ca_s = 0.6$, $t = 3.0$. (c) $Ca_s = 0.6$, $t = 6.0$.

Figure 41: Pressure snapshots and elevation for $Ca_s = 0.6$.



(a) $Ca_s = 0.18$, $t = 1.0$. (b) $Ca_s = 0.18$, $t = 3.0$. (c) $Ca_s = 0.18$, $t = 6.0$.

Figure 42: Pressure snapshots and elevation for $Ca_s = 0.18$.

6. Conclusion

In this paper, we have introduced a 3D unfitted Nitsche-XFEM method for incompressible fluid-structure interaction problems involving immersed thin-walled solids. The proposed results extend the work reported in Alauzet et al. (2016) for the 2D case. The key features of the proposed method are the following:

- Discrete strong and weak discontinuities included in a low order (affine) fluid approximation supported by overlapping meshes, which guarantees interfacial mass conservation and optimal accuracy;
- Cut-FEM methodology for consistency. Integration over general polyhedra enabled via a robust and efficient intersection sub-tessellation algorithm for general unstructured (tetrahedral/triangular) meshes;
- Novel approach based on a fictitious interface for handling front elements (tip elements in 2D) in the case of partially intersected fluid domains;
- Consistent Nitsche treatment of the interface coupling (Lagrange multipliers free) and suitable stabilization for robustness.

The comprehensive numerical study reported in Section 5 highlighted the good performance and robustness of the proposed methodology. As a result, the presented method can be considered as a robust and accurate approach to simulate FSI problems with immersed thin-walled solids. Ongoing work focuses on:

- Loosely coupled schemes which avoid the strong coupling of Algorithm 1: with the purpose of reducing computational complexity without compromising stability and accuracy;
- Fluid-structure interaction with solid contact: a particularly delicate problem arises whenever multiple solids intersect the same fluid element.

Data availability

Data will be made available on request.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: M.A. Fernández reports financial support was provided by the French National Research Agency (ANR).

Acknowledgement

F.M. Gerosa, D. Corti and M.A. Fernández were partially supported by the French National Research Agency (ANR), through the SIMR project (ANR-19-CE45-0020).

References

Alauzet, F., 2014. A changing-topology moving mesh technique for large displacements. Eng. Comput. 30, 175–200.

Alauzet, F., Fabrèges, B., Fernández, M.A., Landajuela, M., 2016. Nitsche-xfem for the coupling of an incompressible fluid with immersed thin-walled structures. Computer Methods in Applied Mechanics and Engineering 301, 300–335.

Alauzet, F., Mehrenberger, M., 2010. P1-conservative solution interpolation on unstructured triangular meshes. International Journal for Numerical Methods in Engineering 84, 1552–1588.

Antonietti, P., Houston, P., Pennesi, G., 2018. Fast numerical integration on polytopic meshes with applications to discontinuous galerkin finite element methods. J. Sci. Comput. 77, 1339–1370.

Astorino, M., Gerbeau, J.F., Pantz, O., Traore, K.F., 2009. Fluid–structure interaction and multi-body contact: application to aortic valves. Computer Methods in Applied Mechanics and Engineering 198, 3603–3612.

Baaijens, F., 2001. A fictitious domain/mortar element method for fluid-structure interaction. Int. Jour. Num. Meth. Fluids 35, 743–761.

Becker, R., Burman, E., Hansbo, P., 2009. A Nitsche extended finite element method for incompressible elasticity with discontinuous modulus of elasticity. Comput. Methods Appl. Mech. Engrg. 198, 3352–3360.

Bergmann, M., Fondanèche, A., Iollo, A., 2022. An Eulerian finite-volume approach of fluid-structure interaction problems on quadtree meshes. Journal of Computational Physics 471, 111647.

Boffi, D., Cavallini, N., Gardini, F., Gastaldi, L., 2012a. Local mass conservation of Stokes finite elements. J. Sci. Comput. 52, 383–400.

Boffi, D., Cavallini, N., Gardini, F., Gastaldi, L., 2012b. Stabilized Stokes elements and local mass conservation. Boll. Unione Mat. Ital. (9) 5, 543–573.

Boffi, D., Cavallini, N., Gastaldi, L., 2011. Finite element approach to immersed boundary method with different fluid and solid densities. Math. Models Methods Appl. Sci. 21, 2523–2550.

Boffi, D., Cavallini, N., Gastaldi, L., 2015. The finite element immersed boundary method with distributed lagrange multiplier. SIAM Journal on Numerical Analysis 53, 2584–2604.

Boffi, D., Gastaldi, L., 2017. A fictitious domain approach with lagrange multiplier for fluid-structure interactions. Numerische Mathematik 135, 711–732.

Boilevin-Kayl, L., Fernández, M.A., Gerbeau, J.F., 2019a. A loosely coupled scheme for fictitious domain approximations of fluid-structure interaction problems with immersed thin-walled structures. SIAM J. Sci. Comput. 41, B351–B374.

Boilevin-Kayl, L., Fernández, M.A., Gerbeau, J.F., 2019b. Numerical methods for immersed fsi with thin-walled structures. Computers & Fluids 179, 744–763.

Burman, E., 2010. Ghost penalty. C. R. Math. Acad. Sci. Paris 348, 1217–1220.

Burman, E., Fernández, M., 2007. Continuous interior penalty finite element method for the time-dependent Navier-Stokes equations: space discretization and convergence. Numer. Math. 107, 39–77.

Burman, E., Fernández, M., 2014. An unfitted Nitsche method for incompressible fluid-structure interaction using overlapping meshes. Comput. Methods Appl. Mech. Engrg. 279, 497–514.

Burman, E., Fernández, M., Hansbo, P., 2006. Continuous interior penalty finite element method for Oseen's equations. SIAM J. Numer. Anal. 44, 1248–1274.

Casquero, H., Bona-Casas, C., Gomez, H., 2017. NURBS-based numerical proxies for red blood cells and circulating tumor cells in microscale blood flow. Comput. Methods Appl. Mech. Engrg. 316, 646–667.

Casquero, H., Bona-Casas, C., Toshniwal, D., Hughes, T., Gomez, H., Jessica Zhang, Y., 2021. The divergence-conforming immersed boundary method: Application to vesicle and capsule dynamics. Journal of Computational Physics 425, 109872.

Chapelle, D., Bathe, K., 2011. The Finite Element Analysis of Shells - Fundamentals. Springer.

Chin, E.B., Sukumar, N., 2020. An efficient method to integrate polynomials over polytopes and curved solids. Computer Aided Geometric Design 82, 101914.

De Hart, J., Peters, G., Schreurs, P., Baaijens, F., 2003. A three-dimensional computational analysis of fluid-structure interaction in the aortic valve. Journal of Biomechanics 36, 103–112.

Deparis, S., Discacciati, M., Fourestey, G., Quarteroni, A., 2006. Fluid-structure algorithms based on Steklov-Poincaré operators. Comput. Methods Appl. Mech. Engrg. 195, 5797–5812.

Di Pietro, D., Ern, A., 2012. Mathematical aspects of discontinuous Galerkin methods. volume 69 of *Mathematics & Applications*. Springer, Heidelberg.

Donea, J., Giuliani, S., Halleux, J.P., 1982. An arbitrary lagrangian-eulerian finite element method for transient dynamic fluid-structure interactions. Computer methods in applied mechanics and engineering 33, 689–723.

Dupont, C., Vidrascu, M., Le Tallec, P., Barthès-Biesel, D., Salsac, A.V., 2022. Modelling the fluid–structure interactions of a capsule using a nonlinear thin shell model: Effect of wall thickness. Journal of Fluids and Structures 113, 103658.

Fernández, M., Moubachir, M., 2005. A Newton method using exact Jacobians for solving fluid-structure coupling. Comp. & Struct. 83, 127–142.
Formaggia, L., Nobile, F., 1999. A stability analysis for the arbitrary Lagrangian Eulerian formulation with finite elements. East-West J. Numer. Math. 7, 105–131.
Frey, P., George, P., 2000. Mesh generation. Application to finite elements. Hermès Science Publ., Paris, Oxford.
Frey, P.J., George, P.L., 2007. Mesh generation: application to finite elements. Iste.
Fries, T.P., Zilian, A., 2009. On time integration in the xfem. International Journal for Numerical Methods in Engineering 79, 69 – 93. doi:10.1002/nme.2558.
Galvin, K., Linke, A., Rebholz, L., Wilson, N., 2012. Stabilizing poor mass conservation in incompressible flow problems with large irrotational forcing and application to thermal convection. Comput. Methods Appl. Mech. Engrg. 237/240, 166–176.
George, P.L., Borouchaki, H., Saltel, E., 2003. 'ultimate'robustness in meshing an arbitrary polyhedron. International Journal for Numerical Methods in Engineering 58, 1061–1089.
Gerstenberger, A., Wall, W., 2008. An extended finite element method/Lagrange multiplier based approach for fluid-structure interaction. Comput. Methods Appl. Mech. Engrg. 197, 1699–1714.
Gil, A.J., Arranz Carreño, A., Bonet, J., Hassan, O., 2013. An enhanced Immersed Structural Potential Method for fluid-structure interaction. Journal of Computational Physics 250, 178–205.
Glowinski, R., Pan, T.W., Hesla, T.I., Joseph, D.D., 1999. A distributed lagrange multiplier/fictitious domain method for particulate flows. International Journal of Multiphase Flow 25, 755–794.
Griffith, B., 2012. On the volume conservation of the immersed boundary method. Commun. Comput. Phys. 12, 401–432.
Hachem, E., Feghali, S., Codina, R., Coupez, T., 2013. Immersed stress method for fluid-structure interaction using anisotropic mesh adaptation. Internat. J. Numer. Methods Engrg. 94, 805–825.
Han, J., Peskin, C.S., 2018. Spontaneous oscillation and fluid–structure interaction of cilia. Proceedings of the National Academy of Sciences 115, 4417–4422.
Hansbo, A., Hansbo, P., 2004. A finite element method for the simulation of strong and weak discontinuities in solid mechanics. Comput. Methods Appl. Mech. Engrg. 193, 3523–3540.
Hansbo, P., Hermansson, J., Svedberg, T., 2004. Nitsche's method combined with space-time finite elements for ALE fluid-structure interaction problems. Comput. Methods Appl. Mech. Engrg. 193, 4195–4206.
Hesch, C., Gil, A.J., Arranz Carreño, A., Bonet, J., 2012. On continuum immersed strategies for Fluid-Structure Interaction. Computer Methods in Applied Mechanics and Engineering 247-248, 51–64.
Hoffman, J., Jansson, J., Stöckli, M., 2011. Unified continuum modeling of fluid-structure interaction. Mathematical Models and Methods in Applied Sciences 21, 491–513.
Jansson, J., Degirmenci, N.C., Hoffman, J., 2017. Adaptive unified continuum fem modeling of a 3d fsi benchmark problem. International journal for numerical methods in biomedical engineering 33, e2851.
Kamensky, D., Hsu, M.C., Schillinger, D., Evans, J.A., Aggarwal, A., Bazilevs, Y., Sacks, M.S., Hughes, T.J., 2015. An immersogeometric variational framework for fluid–structure interaction: Application to bioprosthetic heart valves. Computer methods in applied mechanics and engineering 284, 1005–1053.
Küttler, U., Fürster, C., Wall, W.A., 2006. A solution for the incompressibility dilemma in partitioned fluid-structure interaction with pure dirichlet fluid domains. Computational Mechanics 38, 417–429. doi:10.1007/s00466-006-0066-5.
Landajuela, M., Vidrascu, M., Chapelle, D., Fernández, M.A., 2017. Coupling schemes for the fsi forward prediction challenge: comparative study and validation. International journal for numerical methods in biomedical engineering 33, e2813.
Liu, J., Marsden, A.L., 2018. A unified continuum and variational multiscale formulation for fluids, solids, and fluid–structure interaction. Computer methods in applied mechanics and engineering 337, 549–597.
Liu, Y., Liu, W.K., 2006. Rheology of red blood cell aggregation by computer simulation. Journal of Computational Physics 220, 139–154.
Massing, A., Larson, M.G., Logg, A., 2013. Efficient implementation of finite element methods on nonmatching and overlapping meshes in three dimensions. SIAM J. Sci. Comput. 35, C23–C47.
Mayer, U.M., Gerstenberger, A., Wall, W.A., 2009. Interface handling for three-dimensional higher-order xfem-computations in fluid–structure interaction. International journal for numerical methods in engineering 79, 846–869.
Mayer, U.M., Popp, A., Gerstenberger, A., Wall, W.A., 2010. 3d fluid–structure-contact interaction based on a combined xfem fsi and dual mortar contact approach. Computational Mechanics 46, 53–67.
Mittal, R., Iaccarino, G., 2005. Immersed boundary methods. Annual Review of Fluid Mechanics 37, 239–261.
Nakata, T., Liu, H., 2012. A fluid–structure interaction model of insect flight with flexible wings. Journal of Computational Physics 231, 1822–1847.
Nomura, T., Hughes, T., 1992. An arbitray Lagrangian-Eulerian finite element method for interaction of fluid and rigid body. Comput. Methods Appl. Mech. Eng. 95, 115–138.
Peskin, C.S., 2002. The immersed boundary method. Acta numerica 11, 479–517.
Peskin, C.S., Printz, B.F., 1993. Improved volume conservation in the computation of flows with immersed elastic boundaries. J. Comput. Phys. 105, 33–46.
Sawada, T., Tezuka, A., 2011. LLM and X-FEM based interface modeling of fluid-thin structure interactions on a non-interface-fitted mesh. Comput. Mech. 48, 319–332.
Shahmiri, S., Gerstenberger, A., Wall, W.A., 2011. An xfem-based embedding mesh technique for incompressible viscous flows. International Journal for Numerical Methods in Fluids 65, 166–190.
Stein, K., Tezduyar, T., Benney, R., 2003. Mesh moving techniques for fluid-structure interactions with large displacements. J. Appl. Mech. 70, 58–63.
Strychalski, W., Guy, R.D., 2016. Intracellular pressure dynamics in blebbing cells. Biophys. J. 110, 1168–1179.
Sudhakar, Y., Moitinho de Almeida, J., Wall, W., 2014. An accurate, robust, and easy-to-implement method for integration over arbitrary polyhedra: Application to embedded interface methods. J. Comput. Phys. 273, 393–415.
Takizawa, K., Bazilevs, Y., Tezduyar, T.E., 2012. Space–time and ale-vms techniques for patient-specific cardiovascular fluid–structure interaction modeling. Archives of Computational Methods in Engineering 19, 171–225.

Takizawa, K., Tezduyar, T., Buscher, A., Asada, S., 2014. Space-time interface-tracking with topology change (ST-TC). Comput. Mech. 54, 955–971.

Takizawa, K., Tezduyar, T.E., 2012. Computational methods for parachute fluid–structure interactions. Archives of Computational Methods in Engineering 19, 125–169.

Van Loon, R., Anderson, P.D., Baaijens, F.P., Van de Vosse, F.N., 2005. A three-dimensional fluid–structure interaction method for heart valve modelling. Comptes Rendus Mecanique 333, 856–866.

Van Loon, R., Anderson, P.D., De Hart, J., Baaijens, F.P., 2004. A combined fictitious domain/adaptive meshing method for fluid–structure interaction in heart valves. International Journal for Numerical Methods in Fluids 46, 533–544.

Wang, X., Zhang, L.T., 2013. Modified immersed finite element method for fully-coupled fluid–structure interactions. Computer methods in applied mechanics and engineering 267, 150–169.

Weymouth, G.D., Dommermuth, D.G., Hendrickson, K., Yue, D.K.P., 2006. Advancements in cartesian-grid methods for computational ship hydrodynamics, in: 26th Symposium on Naval Hydrodynamics (16/09/06 - 21/09/06), Rome, Italy.

Wick, T., 2013. Flapping and contact fsi computations with the fluid-solid interface-tracking/interface-capturing technique and mesh adaptivity. Computational Mechanics 53. doi:10.1007/s00466-013-0890-3.

Zhang, L., Gerstenberger, A., Wang, X., Liu, W., 2004. Immersed finite element method. Comput. Methods Appl. Mech. Engrg. 193, 2051–2067.

Zilian, A., Legay, A., 2008. The enriched space-time finite element method (EST) for simultaneous solution of fluid-structure interaction. Internat. J. Numer. Methods Engrg. 75, 305–334.

Zonca, S., Vergara, C., Formaggia, L., 2018. An unfitted formulation for the interaction of an incompressible fluid with a thick structure via an XFEM/DG approach. SIAM Journal on Scientific Computing 40, B59–B84.