Applied Mathematics and Computation 333 (2018) 376–400

Contents lists available at ScienceDirect

Applied Mathematics and Computation

journal homepage: www.elsevier.com/locate/amc

Implicit finite element methodology for the numerical modeling of incompressible two-fluid flows with moving hyperelastic interface

Aymen Laadhari

Institut für Bildverarbeitung, Department of Information Technology and Electrical Engineering, Swiss Federal Institute of Technology Zürich (Eidgenössische Technische Hochschule Zürich, ETH-Zentrum), Zürich CH-8092, Switzerland

ARTICLE INFO

Keywords:
Fluid-structure interaction
Hyperelastic membrane
Incompressible flow
Newton
Monolithic
Finite element method

ABSTRACT

We present a numerical methodology based on the use of the Newton and level set methods and tailored for the simulation of incompressible immiscible two-fluid flows with moving hyperelastic membrane. The method features the use of implicit time integration schemes and is based on a consistent Newton–Raphson linearization. The performances are enhanced by using the Kou's method (Kou et al., 2006) which features a third-order convergence behavior without requiring higher order derivatives. To overcome numerical instability issues related to the explicit decoupling, a fully monolithic strategy and a partitioned implicit strategy are devised. We investigate the main features of the proposed strategies, and we report several numerical experiments with the aim of illustrating their robustness and accuracy. We show numerically that the monolithic strategy performs better and remains stable when considering relatively small viscosities or large stiffness, for which the partitioned approach depicts a slow convergence or even fails to converge. However, the partitioned strategy features significant computational savings when it converges within a reasonable number of sub-iterations.

1. Introduction

This computational framework is concerned with the modeling of thin elastic structures immersed in incompressible Newtonian flows. The coupled dynamics of fluids and lightweight elastic membranes with small thickness compared to the membrane dimensions is of significant interest in many real life phenomena and industrial applications, including amongst which biomedicine, aeronautics and structural engineering. In recent years, this problem has generated a lot of interest from different scientific communities, and researches have covered the fields of theoretical biology [2], mechanics [3], physics [4,5], biomedical engineering [6] and applied mathematics [7].

Among a broad spectrum, let us first list two applications we have in mind for this kind of problems. (i) A cardiovascular application is the dynamics of the aortic valve at the exit of the left ventricle. It is composed of three leaflets that undergo fast movement and large deformations during the opening and closing phases, allowing thereby the correct blood ejection and preventing regurgitation [8]. Typically, the leaflets thickness is very small (about 0.5 mm) compared to the dimensions of the valve (height ≈ 18 mm and aorta diameter ≈ 26 mm) [9,10]. (ii) A second target application is the dynamics of biomembranes such as red blood cells (RBCs) and capsules. A RBC has approximately the dimension $8 \times 2\,\mu$m

E-mail addresses: Aymen.Laadhari@math.ch, laadhari@vision.ee.ethz.ch

https://doi.org/10.1016/j.amc.2018.03.074
0096-3003/

and consists of thin lipid bilayer lacking nucleus and organelles with 4/5 nm-thickness juxtaposing a filamentous meshwork of proteins called cytoskeleton of about 60/80 nm-thickness and enclosing a hemoglobin fluid [5,7,11,12]. The RBC's membrane can be considered as a two-dimensional structure suspended in blood and endowed with hyperelastic properties. Unlike RBCs, capsules are deformable phospholipid membranes with a small resistance to stretching. They are found in nature or are synthesized e.g. for cosmetic, industrial, biomedical and pharmaceutical applications on grounds of their mechanical properties and simple structures. In biomedical applications, they are manufactured to allow drug delivery in the body. When suspended in flow, RBCs and capsules are exposed to viscous stresses that can lead to large deformations of the membrane. It is fundamental to model their mechanical properties, as they enable to establish the resistance of the membrane to the applied stresses.

A fluid-structure interaction problem involving a thick structure requires the simultaneous solution of the fluid and structure problems in such a way that the coupling is achieved by enforcing the kinematic conditions describing the continuity of the velocity and the dynamic conditions describing the balance of stresses at the interface. Further difficulties are encountered when extremely thin structures (membranes) are involved. Throughout the literature, the existing methods can be classified based on the framework used to represent the structure in the fluid domain and how the coupling is performed. The most popular approaches are the Arbitrary Lagrangian Eulerian (ALE) methods and the Immersed Boundary Methods (IBM) [13]. (i) The standard ALE approaches consist in the employment of a body-fitted moving mesh that follows the structural deformations. The fluid and structural meshes remain contiguous, while the compatibility conditions are applied at the shared interface. Different Lagrangian and Eulerian representations are traditionally used for the solid and fluid problems, respectively. ALE-type approaches have been successfully employed in problems with relatively small structural deformations. Efficient and robust remeshing tools are required to maintain high quality mesh elements during the simulation, leading thereby to a substantial computational burden. Major concerns are additionally related to the large structural deformations and the multitude of interpolations between different frameworks. (ii) To address the remeshing issues, several approaches have been developed using fixed grids such as the IBM extensively used in this context [14]. The driving idea is that a Lagrangian description is used for the elastic structure (nonconforming mesh), while keeping a separate Eulerian model for the fluid [15–17]. A common domain is then occupied by both the fluid and structure, while additional forcing (i.e. body or singular surface force) is applied to the fluid problem in the surrounding of the solid domain. Several IB-type approaches have been devised, see e.g. [18–22], the finite difference method [23], the immersed finite element method [24], and the fictitious domain method [25]. We further cite other available strategies using the finite element method [12,26,27] the boundary-element method [5,28,29], and the extended finite element method [30]. The IB-type methods seem to be more appropriate than ALE methods when dealing with large structural deformations. However, an appropriate formulation of the solid mechanics and the coupling conditions is required, while the price to be paid is the loss of accuracy due to the regularization procedure [20,31]. They also allow for semi-implicit time schemes at the price of a stability condition that can be severe depending on the physical parameters of the problem [31,32].

The convergence and stability can be a big concern and relies importantly on the techniques used to enforce the coupling conditions. We can distinguish between monolithic and partitioned approaches. (i) Monolithic approaches take into account the coupling into a single system, which is solved using a single solver [33]. (ii) Partitioned approaches solve the two-way fluid/solid coupled problem in a sequential manner. Hence, separate dedicated solvers involving different frameworks for the fluid and solid are used, and the coupling is performed at the interface level. The advantage of this approach is its modularity and the possibility of re-using standard solvers. (ii-1) Performing one or few iterations at each time step leads to a loosely-coupled approach and an explicit decoupling scheme. That can exhibit numerical instabilities due to the artificial added mass effect which results from the effect of an extra mass the fluid force exert on the structural interface degrees of freedom [34]. The numerical difficulties are exacerbated for problems involving extremely slender and lightweight structures in an incompressible fluid [35]. (ii-2) To circumvent instabilities inherent in loosely-coupled schemes, the strongly-coupled approaches represent somewhat a remedy that requires an iterative process at each time level until the convergence is achieved.

The central aim of the present paper is to present a stable computational framework to solve problems involving an elastic membrane in an incompressible fluid. The membrane is tracked using a level set method. This Eulerian framework allows a unified representation and a unique solver for the membrane and fluid. To address stability issues due to the time step restrictions when using explicit methods, we present fully implicit strategies based on the use of an exact Newton method. To improve the convergence properties, we consider a generalization to the multidimensional case of the Kou's variant [1,36], which features a faster convergence behavior without computing higher order derivatives. To our knowledge, such Newton variants have not been investigated for the present problem. We propose both a fully monolithic approach and a strongly-coupled approach, and we investigate their stability properties and the corresponding computational costs.

We have arranged the remainder of the paper as follows. Section 2 presents the mathematical formulation of the problem. Section 3 focuses on the numerical method and describes in detail the fully monolithic and the strongly coupled partitioned approaches. The two approaches use the cubically-convergent Kou's method. In Section 4, we report several numerical examples to illustrate the efficiency and the robustness of the proposed methods. We close with some comments about the model limitations and forthcoming extensions in Section 5.

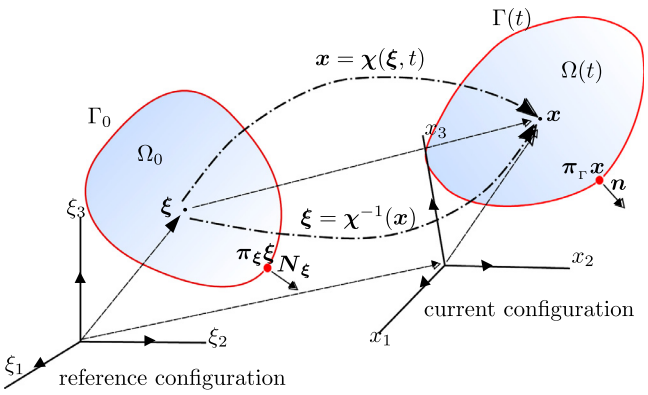


Fig. 1. Schematic representation of a thin elastic membrane in the reference and current configurations.

2. Mathematical formulation

2.1. Membrane elasticity and constitutive equations

Let $T>0$ be the period of the experiment. For any time $t\in(0,\ T)$, let $\Gamma(t)$ represent an elastic membrane. We use the shorthand notation $\mathbf{sym}(\boldsymbol{T}) = \boldsymbol{T} + \boldsymbol{T}^T$ to denote the symmetric part of any tensor $\boldsymbol{T}$.

We focus on both deformation and motion of a thin elastic membrane under the effect of viscous fluid forces. To follow the membrane deformations in an Eulerian manner, we need an elastic constitutive equation that expresses the membrane stress tensor in the same current frame as the fluid problem. The mathematical formulation in the case of an elastic membrane is different from that in the case of a thick elastic structure. We first briefly recall the fundamentals of the Gurtin–Murdoch surface elasticity theory [37–39], required to elaborate the final model.

Let us consider a domain in two different configurations, see Fig. 1. At time $t=0$, the reference configuration $\Omega_0\subset\mathbb{R}^d, d=2,3$, has an elastic boundary Γ_0. The domain and membrane are represented by Ω and Γ in the current configuration at time $t>0$, such that $\boldsymbol{x}=\boldsymbol{\chi}(\boldsymbol{\xi},t)$ is the current position of a material particle having the position $\boldsymbol{\xi}$ in Ω_0 and $\boldsymbol{\chi}$: $\Omega_0\to(0,\ T)\times\Omega$ is a smooth one-to-one deformation map. Let $\boldsymbol{\nabla}_{\boldsymbol{\xi}}$ be the material gradient, whereas $\boldsymbol{\nabla}$ and div stand for the gradient and divergence operators with respect to spatial coordinates. We denote by $\mathbf{N}_{\boldsymbol{\xi}}$ and $\mathbf{n}$ the outward pointing normals on the membrane in the reference and current configurations, respectively. Let the symbol $\otimes$ denote the outer product and $\mathbf{I}$ be the identity tensor. The surface projectors: $\boldsymbol{\pi}_{\boldsymbol{\xi}} = \mathbf{I} - \mathbf{N}_{\boldsymbol{\xi}}\otimes\mathbf{N}_{\boldsymbol{\xi}}$ and $\boldsymbol{\pi}_\Gamma = \mathbf{I}-\mathbf{n}\otimes\mathbf{n}$ project any vector or tensor onto the tangent planes to Γ_0 and $\Gamma(t)$, respectively. Therefore, the surface gradient $\boldsymbol{\nabla}_s = \boldsymbol{\pi}_\Gamma\ \boldsymbol{\nabla}$ and surface divergence $div_s = \mathrm{tr}\big(\boldsymbol{\nabla}_s\big)$ operators are defined in the current configuration. The deformation gradient tensor $\mathbf{F}$ represents the Jacobian matrix of the deformation mapping and reads $\mathbf{F} = \boldsymbol{\nabla}_{\boldsymbol{\xi}}\boldsymbol{\chi} = \boldsymbol{\nabla}\boldsymbol{\xi}^{-1}$ using Lagrangian and Eulerian descriptions, respectively, as a consequence of the relation $\boldsymbol{\chi}^{-1}(\boldsymbol{\chi}(\boldsymbol{\xi},t)) = \boldsymbol{\xi}$. Let $\boldsymbol{u}(\boldsymbol{x},\ t)\equiv\partial_t\boldsymbol{\chi}$ be the spatial velocity. The deformation gradient tensor and the inverse deformation map $\boldsymbol{\chi}^{-1}$ (backward characteristics) verify:

$$\partial_t\mathbf{F} + \boldsymbol{u}\cdot\boldsymbol{\nabla}\mathbf{F} = \boldsymbol{\nabla}\boldsymbol{u}\,\mathbf{F} \quad \text{and} \quad \partial_t\boldsymbol{\xi} + \boldsymbol{u}\cdot\boldsymbol{\nabla}\boldsymbol{\xi} = \mathbf{0} \tag{2.1}$$

allowing consequently a fully Eulerian description of the motion of thick structures, see e.g. [27]. At the numerical level, the advection of $\boldsymbol{\chi}^{-1}$ is not always a good choice for the Eulerian description. Indeed, several problems may arise in the case of large deformations when $\det(\boldsymbol{\nabla}\xi)$ locally vanishes, see [27].

To introduce the elasticity of an extremely thin membrane in an Eulerian framework, we first express the elasticity problem in terms of the surface deformation gradient such that we lack all out-of-plane strain components. To track the positions of material points on the membrane, we introduce the surface deformation gradient $\mathbf{F}_s = \boldsymbol{\nabla}_{\boldsymbol{\xi}}\boldsymbol{x}|_\Gamma$ which writes $\mathbf{F}_s = \boldsymbol{\pi}_\Gamma\mathbf{F}\boldsymbol{\pi}_{\boldsymbol{\xi}}$ [38]. Note that $\mathbf{n}.\mathbf{F}_s = \mathbf{0}$ (two-dimensionality assumption) and $\mathbf{F}_s.\mathbf{N} = \mathbf{0}$ which means that the tensor $\mathbf{F}_s$ is rank deficient with a zero eigenvalue corresponding to the eigenvector $\mathbf{N}$. Indeed, the tensor $\mathbf{F}_s$ has the same dimension as $\mathbf{F}$ but only describes the in-plane deformations in the tangential space. The left Cauchy–Green surface strain tensor $\mathbf{B}_s$ and the stretch tensor $\boldsymbol{\Lambda}$ are introduced such that $\mathbf{B}_s = \boldsymbol{\Lambda}^2 = \mathbf{F}_s\mathbf{F}_s^T = \boldsymbol{\pi}_\Gamma\hat{\mathbf{B}}_s\boldsymbol{\pi}_\Gamma$ with $\hat{\mathbf{B}}_s = \mathbf{F}\boldsymbol{\pi}_{\boldsymbol{\xi}}\mathbf{F}^T$.

The eigenvalues λ_1 and λ_2 of $\boldsymbol{\Lambda}$ correspond, respectively, to the eigenvectors $\mathbf{e}_1$ and $\mathbf{e}_2$ defining the local tangent plane to the membrane. They represent the principal extension ratios of the membrane and allow the quantification of the deformations in terms of both stretch and shear. It is convenient to introduce the surface strain invariants:

$$\mathcal{I}_1^s \equiv \ln(\lambda_1\lambda_2) = \frac{1}{2}\Big(\ln\big((\mathrm{tr}\mathbf{B}_s)^2 - \mathrm{tr}\mathbf{B}_s^2\big) - \ln 2\Big) \quad \text{and} \quad \mathcal{I}_2^s \equiv \frac{1}{2}\big(\lambda_1^2+\lambda_2^2-2\big) = \frac{1}{2}\mathrm{tr}\mathbf{B}_s - 1.$$

The ratio of the deformed to the undeformed surface area $\mathcal{J}_s \equiv \lambda_1\lambda_2 = \exp(\mathcal{I}_1^s)$ represents the change in the local membrane area. The strain energy refers to the energy stored in the deformed structure. Analogously to the bulk case, the mechanical response of a hyperelastic membrane is described by a scalar stored energy that depends on the surface deformation

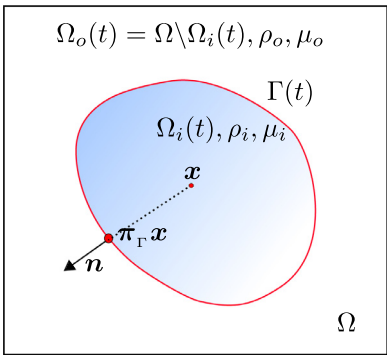


Fig. 2. Sketch for the thin elastic membrane and its surrounding fluid domain.

tensor $\mathbf{F}_s$. That means that the stored energy density $\mathcal{W}$ is solely expressed with respect to the invariants of $\mathbf{F}_s$. Under the transversely isotropic assumption of a very thin sheet, we suppose that the membrane obeys a Neo-Hookean constitutive law [38]. The surface strain energy is defined as follows:

$$\mathcal{W}(\mathbf{F}_s) = \mathcal{W}(\mathcal{I}_1^s, \mathcal{I}_2^s) = \frac{\mathsf{E}_s}{6}\left(\lambda_1^2 + \lambda_2^2 + \frac{1}{(\lambda_1\lambda_2)^2} - 3\right) = \frac{\mathsf{E}_s}{6}\left(2\mathcal{I}_2^s + \exp\left(-2\mathcal{I}_1^s\right) - 1\right),$$

where E_s is the membrane elastic modulus. Remark that the surface strain energy $\mathcal{W}$ is symmetric with respect to λ_1 and λ_2. Analogously to the bulk case, the surface Cauchy–Green stress $\boldsymbol{\sigma}_s$ can be expressed with respect to the first surface Piola Kirchhoff stress tensor. Since the surface deformation gradient is rank deficient, we introduce the inverse surface deformation gradient via the relation: $\hat{\mathbf{F}}_s \equiv \nabla\boldsymbol{\xi}|_{\Gamma_0}$. That leads to the relations $\mathbf{F}_s\hat{\mathbf{F}}_s = \boldsymbol{\pi}_\Gamma$ and $\hat{\mathbf{F}}_s\mathbf{F}_s = \boldsymbol{\pi}_{\boldsymbol{\xi}}$. The derivatives of the surface invariants with respect to the surface deformation gradient $\mathbf{F}_s$ read:

$$\frac{\partial \mathcal{I}_1^s}{\partial \mathbf{F}_s} = \hat{\mathbf{F}}_s^T \quad \text{and} \quad \frac{\partial \mathcal{I}_2^s}{\partial \mathbf{F}_s} = \mathbf{F}_s.$$

Accordingly, the membrane Cauchy-stress tensor is given by:

$$\boldsymbol{\sigma}_s = \frac{1}{\mathcal{J}_s}\frac{\partial \mathcal{W}}{\partial \mathbf{F}_s}\mathbf{F}_s^T = \frac{1}{\mathcal{J}_s}\sum_{i=1,2}\frac{\partial \mathcal{W}}{\partial \mathcal{I}_i^s}\frac{\partial \mathcal{I}_i^s}{\partial \mathbf{F}_s}\mathbf{F}_s^T = \frac{1}{\exp\left(\mathcal{I}_1^s\right)}\left(\frac{\partial \mathcal{W}}{\partial \mathcal{I}_1^s}\boldsymbol{\pi}_\Gamma + \frac{\partial \mathcal{W}}{\partial \mathcal{I}_2^s}\mathbf{B}_s\right) = \frac{\mathsf{E}_s}{3\mathcal{J}_s}\left(\mathbf{B}_s - \exp\left(-2\mathcal{I}_1^s\right)\boldsymbol{\pi}_\Gamma\right).$$

To derive the equations governing the membrane deformation in a surrounding fluid, we shall write the balance of elastic and hydrodynamic forces in an infinitesimal section on Γ. In addition, we assume that the membrane follows the movement of the fluid on both sides with the same velocity. Therefore, the resulting elastic force $\mathcal{F}_\Gamma = \mathbf{div}_s\,\boldsymbol{\sigma}_s$ describes the discontinuity of the normal fluid stress tensor across the membrane Γ.

If the mechanics is restricted to the stretching [22,23], the elastic force can readily be deduced from the previous equation. In that case, the stored energy function $\mathcal{W} = \mathcal{W}(\mathcal{J}_s)$ leads to the following expression of the elastic force:

$$\mathcal{F}_\Gamma = \mathbf{div}_s\left(\mathcal{W}'(\exp\left(\mathcal{I}_1^s\right))\boldsymbol{\pi}_\Gamma\right) = \nabla_s\mathcal{W}'(\exp\left(\mathcal{I}_1^s\right)) - \mathcal{W}'(\exp\left(\mathcal{I}_1^s\right))H\,\mathbf{n}, \tag{2.2}$$

where $H = div_s\,\mathbf{n}$ denotes the mean curvature defined along the membrane Γ. The previous expression of the membrane force corresponds to the same expression obtained in by Lai et al. in [22].

Finally, an Eulerian description of the motion has to recover the deformation history everywhere on the membrane. That may be, for instance, described by the advection of $\mathbf{F}_s$. Analogously to the bulk case involving thick structures in [27], we choose to rather advect the surface tensor $\hat{\mathbf{B}}_s$ and formulate the problem in terms of $\hat{\mathbf{B}}_s$. It follows from the advection of the projector tensor and the aforementioned advection equation of $\mathbf{F}$ and its transpose $\mathbf{F}^T$ (2.1) that

$$\partial_t\hat{\mathbf{B}}_s + \boldsymbol{u}\cdot\nabla\hat{\mathbf{B}}_s = \mathbf{sym}(\nabla\boldsymbol{u}\,\hat{\mathbf{B}}_s).$$

Henceforth, the domain Ω stands for the entire computational domain enclosing the incompressible fluid and the membrane in the current configuration, see Fig. 2. For any time $t\in(0,\,T)$, let $\Omega_{i/o}(t) \subset \mathbb{R}^d$ with $d = 2, 3$ denote the internal and external domains, respectively. We assume a Lipschitz continuous membrane $\Gamma(t) = \partial\Omega_i(t)$ such that $\Gamma(t)\cap\partial\Omega = \emptyset$, $\forall t\in(0,\,T)$. Here and throughout, the explicit dependence of Ω_i, Ω_o and Γ from t will be understood.

2.2. Level set formulation

We use an Eulerian framework and we implicitly capture the membrane using the level set method, which identifies Γ as the zero level set of a function φ. A time-dependant partial differential equation describes the motion of Γ:

$$\partial_t\varphi + \boldsymbol{u}\cdot\nabla\varphi = 0, \qquad \text{in } (0,T)\times\Omega. \tag{2.3}$$

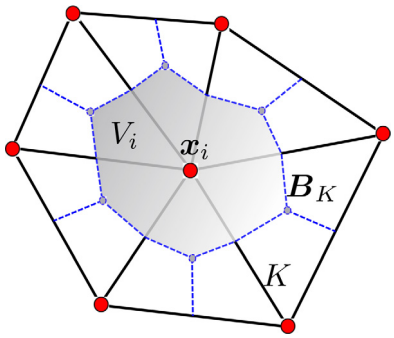


Fig. 3. Schematic diagram of the elementary cell or control volume V_i based on a dual mesh in the two-dimensional case. $\mathbf{B}_K$ is the barycenter of the element $K \in \mathscr{T}_h$, while V_i is centered at the vertex $\boldsymbol{x}_i$.

This problem is initialized with a signed distance function φ_0 corresponding to the initial position of Γ. All the geometrical quantities on the membrane are encoded in terms of φ and are consequently extended to the entire domain Ω. It is not required to use meshes that follow the deformations of the membrane. A regularization approach is commonly introduced using a regularization parameter ε. Let us consider the parameter $\tilde{\varphi} = |\varphi|/\varepsilon$. The Dirac measure δ_Γ needs to be smoothed within a banded strip of width 2ε to obtain a better continuity at the membrane. Following Griffith and Peskin [16], we consider the following expression of the regularized Dirac function $\delta_\varepsilon(\cdot)$:

$$\varepsilon\delta_\varepsilon(\varphi) = \begin{cases} \dfrac{\sqrt{3}}{336}\sqrt{243 + 1584\tilde{\varphi} - 748\tilde{\varphi}^2 - 1560\tilde{\varphi}^3 + 500\tilde{\varphi}^4 + 336\tilde{\varphi}^5 - 112\tilde{\varphi}^6} + \dfrac{61}{112} - \dfrac{11}{42}\tilde{\varphi} - \dfrac{11}{56}\tilde{\varphi}^2 \\ + \dfrac{1}{12}\tilde{\varphi}^3, \ \text{if } |\tilde{\varphi}| < 1 \\ \dfrac{21}{16} + \dfrac{7}{12}\tilde{\varphi} - \dfrac{7}{8}\tilde{\varphi}^2 + \dfrac{1}{6}\tilde{\varphi}^3 - \dfrac{3}{2}\varepsilon\,\delta_\varepsilon(\tilde{\varphi} - 1), & \text{if } 1 \leqslant |\tilde{\varphi}| < 2 \\ \dfrac{9}{8} - \dfrac{23}{12}\tilde{\varphi} + \dfrac{3}{4}\tilde{\varphi}^2 - \dfrac{1}{12}\tilde{\varphi}^3 + \dfrac{1}{2}\varepsilon\,\delta_\varepsilon(\tilde{\varphi} - 2), & \text{if } 2 \leqslant |\tilde{\varphi}| < 3 \\ 0, & \text{otherwise.} \end{cases}$$

We consider the following expressions of the regularized Heaviside function $\mathscr{H}_\varepsilon$ and the regularized sign function $\mathcal{S}_\varepsilon$:

$$\mathscr{H}_\varepsilon(\varphi) = \begin{cases} 0, & \text{if } \varphi < -\varepsilon \\ \dfrac{1}{2}\left(1 + \dfrac{\varphi}{\varepsilon} + \dfrac{1}{\pi}\sin\left(\dfrac{\pi\varphi}{\varepsilon}\right)\right), & \text{if } |\varphi| \leqslant \varepsilon \\ 1, & \text{otherwise} \end{cases} \quad \text{and} \quad \mathcal{S}_\varepsilon(\varphi) = 2\mathscr{H}_\varepsilon(\varphi) - 1.$$

At the numerical level, the regularization parameter ε is set element-wise proportional to the local mesh size. Let us consider a partition $\mathscr{T}_h$ of Ω consisting of geometrically conforming open simplicial elements K (triangles for $d = 2$ and tetrahedra for $d = 3$), such that $\overline{\Omega} = \bigcup_{K\in\mathscr{T}_h}$. Let h_K stand for the average mesh-size of the mesh element $K \in \mathscr{T}_h$. Although almost regular meshes will be generated, we consider a general setting suitable for simulations with local mesh refinements. The case that we have in mind for future work is the use of mesh adaptations to both improve the convergence properties of the present method and accurately capture complex flow patterns in the aortic sinuses [40]. In practice, we set $\varepsilon = 3h_K$, $\forall K \in \mathscr{T}_h$. The local mesh size h_K is evaluated for a general unstructured mesh as follows.

For ease of exposition, we only present the two-dimensional case. We compute h_K using the assembled lumped-mass matrix [41]. Each diagonal entry of this diagonal matrix represents the area of a control volume V_i centered at the vertex $\boldsymbol{x}_i$ and joining the barycenters of the neighboring element to the edges' mid-points as depicted in Fig. 3. Hence, h_K is computed with respect to the square root of this diagonal entries. In what follows, almost regular meshes are built and we consider a global value of the mesh size $h = \max_{K\in\mathscr{T}_h}\text{diam}(K)$ in the convergence analyses.

For any generic function $\eta(\cdot)$ defined on Γ, let $\tilde{\eta}(\cdot)$ represent an extension of $\eta(\cdot)$ to the domain Ω such that it's constant in the normal direction to the membrane. The surface integrals are transformed into integrals over the entire domain Ω and approximated as follows:

$$\int_\Gamma \eta(\boldsymbol{x})\, d\boldsymbol{s} = \int_\Omega |\nabla\varphi|\ \delta_\Gamma\,\tilde{\eta}(\boldsymbol{x})\, d\boldsymbol{x} \approx \int_\Omega |\nabla\varphi|\,\delta_\varepsilon(\varphi)\,\tilde{\eta}(\boldsymbol{x})\, d\boldsymbol{x}.$$

Due to the regularization procedure, the standard finite element approximation would result in a suboptimal convergence behavior. We refer to Section 4.2.2 for a discussion about this issue. The signed distance property is lost after the advection of φ, leading to problematic configurations with too large or too small values of $|\nabla\varphi|$ near the membrane which can

deteriorate the computational accuracy. We regularly solve a redistancing problem to reestablish the signed distance property, while keeping invariant the iso-surface zero without any unphysical shifting [42,43]. To alleviate this difficulty, we use a method that consists in incorporating a forcing term in the original redistancing problem in order to prevent the unphysical shifting during the reinitialization process. The numerical method uses the method of characteristics, and we refer to [44–46] for a detailed description and validation of the numerical method.

2.3. Statement of the nonlinear fluid-membrane interaction problem

We state the equations governing the deformations of the elastic thin structure and the coupling with the fluid dynamics. We assume the fluid to be Newtonian, viscous and incompressible in both Ω_i and Ω_o. The flow is governed by the instationary Navier–Stokes equations. We consider piecewise constant fluid density $\rho_{i/o}$ and dynamic viscosity $\mu_{i/o}$ in the inner and outer domains, respectively. The regularized viscosity and density functions are given by $\mu_\varepsilon(\varphi) = \mu_i + (\mu_o - \mu_i)\mathscr{H}_\varepsilon(\varphi)$ and $\rho_\varepsilon(\varphi) = \rho_i + (\rho_o - \rho_i)\mathscr{H}_\varepsilon(\varphi)$, respectively. Let $\mathbf{D}(\boldsymbol{u}) = \mathbf{sym}(\nabla \boldsymbol{u})/2$ and $\boldsymbol{\sigma}_f = 2\mu(\varphi)\mathbf{D}(\boldsymbol{u}) - p\mathbf{I}$ represent the rate of deformation tensor and the fluid Cauchy stress tensor for Newtonian fluids, respectively. Let $[.]^+_-$ design the jumps in velocity and normal stress across the membrane. The interfacial conditions require a continuous velocity across the membrane. From the differential balance of forces on an infinitesimal portion of the membrane, the hydrodynamic traction undergoes a discontinuity due to the membrane mechanics, and the stress discontinuity across Γ expresses in terms of the elastic force $\mathcal{F}_\Gamma$ as $[\boldsymbol{\sigma}_f.\mathbf{n}]^+_- = -\mathcal{F}_\Gamma$. We also consider an external body force, referred to as $\boldsymbol{g}$. Collecting the elements above, we obtain the following fluid-structure interaction problem:

$\mathscr{P}$: find the velocity $\boldsymbol{u}$, pressure p, surface tensor $\hat{\mathbf{B}}_s$ and level set function φ such that

$$\partial_t \varphi + \boldsymbol{u}\cdot\nabla\varphi = 0, \quad \text{in } (0,T)\times\Omega, \tag{2.4a}$$

$$\rho_\varepsilon(\varphi)(\partial_t \boldsymbol{u} + \boldsymbol{u}.\nabla\boldsymbol{u}) - \mathbf{div}(2\mu_\varepsilon(\varphi)\mathbf{D}(\boldsymbol{u})) + \nabla p = \rho_\varepsilon(\varphi)\boldsymbol{g} \quad \text{in } (0,T)\times\Omega, \tag{2.4b}$$

$$div\ \boldsymbol{u} = 0 \quad \text{in } (0,T)\times\Omega, \tag{2.4c}$$

$$[\boldsymbol{u}]^+_- = \mathbf{0} \quad \text{on } (0,T)\times\Gamma, \tag{2.4d}$$

$$[\boldsymbol{\sigma}_f\mathbf{n}]^+_- = -\mathbf{div}_s\,\boldsymbol{\sigma}_s \quad \text{on } (0,T)\times\Gamma, \tag{2.4e}$$

$$\partial_t\hat{\mathbf{B}}_s + \boldsymbol{u}\cdot\nabla\hat{\mathbf{B}}_s = \mathbf{sym}(\nabla\boldsymbol{u}\,\hat{\mathbf{B}}_s) \quad \text{in } (0,T)\times\Omega, \tag{2.4f}$$

where suitable initial and boundary conditions are considered. The stress jump across Γ leads to a singular elastic force localized on Γ. In a variational manner, let $\boldsymbol{v}$ be a test function corresponding to the velocity $\boldsymbol{u}$. Assuming a closed membrane, the boundary integral term on $\partial\Gamma$ resulting from the Green transformation for surface integrals vanishes. From the symmetry and two-dimensionality assumptions of the elastic response, the force term involving the membrane tension tensor reads:

$$\int_\Gamma \mathbf{div}_s\,\boldsymbol{\sigma}_s\cdot\boldsymbol{v} = -\int_\Gamma \boldsymbol{\sigma}_s : \nabla_s\boldsymbol{v} + \int_\Gamma H(\boldsymbol{\sigma}_s.\mathbf{n}).\boldsymbol{v} = -\int_\Gamma \boldsymbol{\sigma}_s : \left(\boldsymbol{\pi}_\Gamma\frac{\nabla_s\boldsymbol{v}+\nabla_s\boldsymbol{v}^T}{2}\boldsymbol{\pi}_\Gamma\right) = \int_\Gamma \frac{\mathsf{E}_s}{3\mathcal{J}_s^3}\,div_s\,\boldsymbol{v} - \frac{\mathsf{E}_s}{3\mathcal{J}_s}\hat{\mathbf{B}}_s : \boldsymbol{\pi}_\Gamma.\nabla\boldsymbol{v}.\boldsymbol{\pi}_\Gamma.$$

To solely express the invariants with respect to the tensor $\hat{\mathbf{B}}_s$, we show that:

$$\mathrm{tr}(\mathbf{B}_s) = \mathbf{B}_s : \mathbf{I} = \hat{\mathbf{B}}_s : \boldsymbol{\pi}_\Gamma \quad \text{and} \quad \mathrm{tr}\left(\mathbf{B}_s^2\right) = \mathbf{B}_s^2 : \mathbf{I} = \mathbf{B}_s : \mathbf{B}_s = \hat{\mathbf{B}}_s\boldsymbol{\pi}_\Gamma\hat{\mathbf{B}}_s : \boldsymbol{\pi}_\Gamma = (\hat{\mathbf{B}}_s\boldsymbol{\pi}_\Gamma)^2 : \mathbf{I}.$$

By expressing the projector tensor $\boldsymbol{\pi}_\Gamma$ in the local basis $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{n})$ associated to the tangent plane to the membrane at any position on the membrane, we obtain $\mathrm{tr}(\boldsymbol{\pi}_\Gamma) = \boldsymbol{\pi}_\Gamma : \mathbf{I} = 2$. Therefore, the strain invariants can be expressed as:

$$\mathcal{J}_s = \sqrt{\frac{1}{2}\left(\left(\hat{\mathbf{B}}_s\boldsymbol{\pi}_\Gamma : \mathbf{I}\right)^2 - \left(\hat{\mathbf{B}}_s\boldsymbol{\pi}_\Gamma\right)^2 : \mathbf{I}\right)} \qquad \text{and} \qquad I_2^s = \tfrac{1}{2}\left(\hat{\mathbf{B}}_s - \mathbf{I}\right) : \boldsymbol{\pi}_\Gamma.$$

In the particular two-dimensional case, only one principal planar stretch exists in the tangent direction $\nabla\wedge\varphi/|\nabla\varphi|$. It helps to describe the change on length between the reference and actual configuration. That results in $\mathcal{J}_s^2 = \mathrm{tr}(\mathbf{B}_s) = \hat{\mathbf{B}}_s : \boldsymbol{\pi}_\Gamma$. Therefore, the initial condition $\hat{\mathbf{B}}_s(t=0) = \varsigma_0^2\boldsymbol{\pi}_\Gamma$ allows prescribing a uniform initial stretching along the membrane $\mathcal{J}_s(t=0) = \varsigma_0$.

Let $\Sigma_- = \{\boldsymbol{x} \in \partial\Omega : \boldsymbol{u}\cdot\boldsymbol{\nu}(\boldsymbol{x}) < 0\}$ be the upstream boundary. Before stating an appropriate weak formulation, we introduce the following function spaces:

$$\mathbb{V}(\boldsymbol{u}_b) = \left\{\boldsymbol{v} \in \left(H^1(\Omega)\right)^d : \boldsymbol{v} = \boldsymbol{u}_b \text{ on } \Sigma_D\right\}, \quad \mathbb{Q} = \left\{q \in L^2(\Omega) : \int_\Omega q\, d\boldsymbol{x} = 0\right\},$$
$$\mathbb{W} = \{\boldsymbol{\tau} \in L^2(\Omega)^{d\times d} : \boldsymbol{\tau} = \boldsymbol{\tau}^T\}, \quad \text{and} \quad \mathbb{X}(\varphi_b) = \{\psi \in W^{1,\infty}(\Omega)\cap H^1(\Omega) : \ \psi = \varphi_b \text{ on } \Sigma_-\}.$$

In what follows, we only describe the numerical methodology in the general three-dimensional case.

3. Numerical strategy

In this section, we present two implicit strategies to solve the aforementioned problem. To devise an implicit scheme, the partitioned Strategy I uses a sub-iteration approach yielding a strongly-coupled strategy. Strategy II is fully monolithic and the coupled system is solved simultaneously in a block-wise manner. The numerical resolution uses a third-order convergent Newton variant without the evaluation of higher-order derivatives.

3.1. Strategy I: partitioned strongly-coupled time scheme formulation

For ease of exposition, we first describe a semi-implicit scheme and the corresponding variational formulation. Thereafter, Strategy I is presented and is based on an iteration procedure.

3.1.1. Time advancing scheme

Let us divide $[0, T]$ into N subintervals $[t^n, t^{n+1})$, $n = 0, \dots, N-1$ of constant size Δt. For any $n \geq 1$, the unknowns $\boldsymbol{u}^n$, p^n, $\hat{\mathbf{B}}_s^n$ and φ^n are computed iteratively at time t^n using values at previous time steps. The notation $\boldsymbol{\pi}_\Gamma^n$ stands for the discretized surface projector tensor $\mathbf{I} - \mathbf{n}^n \otimes \mathbf{n}^n$ at time t^n. Accordingly, we introduce the discretized surface operators $div_s^n(\cdot) = \boldsymbol{\pi}_\Gamma^n : \boldsymbol{\nabla}(\cdot)$ and $\boldsymbol{\nabla}_s^n(\cdot) = \boldsymbol{\pi}_\Gamma^n \boldsymbol{\nabla}(\cdot)$ at t^n. We consider the backward differentiation scheme of second order in (2.4b)–(2.4f). The scheme is bootstrapped by $\boldsymbol{u}^{-1} = \boldsymbol{u}^0$, where $\boldsymbol{u}^{-1}$ only represents a convenient notation, and analogously for $\hat{\mathbf{B}}_s$ and φ. The time derivative terms of $\boldsymbol{u}$ and $\hat{\mathbf{B}}_s$ are approximated by:

$$\frac{\partial \boldsymbol{u}}{\partial t} \approx \frac{3\boldsymbol{u}^n - 4\boldsymbol{u}^{n-1} + \boldsymbol{u}^{n-2}}{2\Delta t} \quad \text{and} \quad \frac{\partial \hat{\mathbf{B}}_s}{\partial t} \approx \frac{3\hat{\mathbf{B}}_s^n - 4\hat{\mathbf{B}}_s^{n-1} + \hat{\mathbf{B}}_s^{n-2}}{2\Delta t}.$$

Regarding the discretization of the material derivative in (2.4a), we use a second order combined characteristic and finite difference discretization method [47]. Let $\mathsf{X}(\cdot, \boldsymbol{x}, t)$ be the characteristic curve passing at time t through $\boldsymbol{x} \in \Omega$, and $\tilde{\boldsymbol{u}} = 2\boldsymbol{u}^n - \boldsymbol{u}^{n-1}$ be the second order extrapolation of the velocity. The first and second-order characteristics are given by: $\mathsf{X}_{n-1}^{(1)}(\boldsymbol{x}) = \boldsymbol{x} - \Delta t\, \tilde{\boldsymbol{u}}(\boldsymbol{x})$ and $\mathsf{X}_{n-2}^{(2)}(\boldsymbol{x}) = \boldsymbol{x} - 2\Delta t\, \tilde{\boldsymbol{u}}(\boldsymbol{x})$, respectively. The material derivative is approximated by:

$$\frac{\partial}{\partial \tau}\left(\varphi(\mathsf{X}(t^n, \boldsymbol{x}; \tau), \tau)\right)\Big|_{\tau = t^n} \approx \frac{3\varphi(t^n, \boldsymbol{x}) - 4\varphi\left(t^{n-1}, \mathsf{X}_{n-1}^{(1)}(\boldsymbol{x})\right) + \varphi\left(t^{n-2}, \mathsf{X}_{n-2}^{(2)}(\boldsymbol{x})\right)}{2\Delta t}.$$

First, let us introduce the following semi-implicit approximation in time of the problem $\mathscr{P}$: find $(\boldsymbol{u}^n, p^n, \hat{\mathbf{B}}_s^n, \varphi^n) \in \mathbb{V}(\boldsymbol{u}_b) \times \mathbb{Q} \times \mathbb{W} \times \mathbb{X}(\varphi^{n-1})$ such that

$$\begin{aligned}
&\int_\Omega \rho_\varepsilon\left(\varphi^{n-1}\right)\left(\frac{3\boldsymbol{u}^n - 4\boldsymbol{u}^{n-1} + \boldsymbol{u}^{n-2}}{2\Delta t} + \boldsymbol{u}^n \cdot \boldsymbol{\nabla}\boldsymbol{u}^n\right)\cdot \boldsymbol{\nu} + \int_\Omega 2\mu_\varepsilon\left(\varphi^{n-1}\right)\mathbf{D}(\boldsymbol{u}^n) : \mathbf{D}(\boldsymbol{\nu}) \\
&\quad - \frac{\mathsf{E}_s}{3}\int_\Omega \frac{1}{(\mathcal{J}_s^n)^3}\left|\boldsymbol{\nabla}\varphi^{n-1}\right|\delta_\varepsilon\left(\varphi^{n-1}\right) div_s^{n-1}\boldsymbol{\nu} \\
&\quad + \frac{\mathsf{E}_s}{3}\int_\Omega \frac{1}{\mathcal{J}_s^n}\hat{\mathbf{B}}_s^n : \boldsymbol{\pi}_\Gamma^{n-1}\boldsymbol{\nabla}\boldsymbol{\nu}\boldsymbol{\pi}_\Gamma^{n-1}\left|\boldsymbol{\nabla}\varphi^{n-1}\right|\delta_\varepsilon\left(\varphi^{n-1}\right) - \int_\Omega p^n\, div\, \boldsymbol{\nu} = \int_\Omega \rho_\varepsilon\left(\varphi^{n-1}\right)\boldsymbol{g}\cdot\boldsymbol{\nu},
\end{aligned} \tag{3.1a}$$

$$\int_\Omega q\ div\, \boldsymbol{u}^n = 0, \tag{3.1b}$$

$$\int_\Omega \frac{3\hat{\mathbf{B}}_s^n - 4\hat{\mathbf{B}}_s^{n-1} + \hat{\mathbf{B}}_s^{n-2}}{2\Delta t} : \boldsymbol{\tau} + \int_\Omega \left(\boldsymbol{u}^n \cdot \nabla\hat{\mathbf{B}}_s^n\right) : \boldsymbol{\tau} - \int_\Omega \mathbf{sym}(\boldsymbol{\nabla}\boldsymbol{u}^n\hat{\mathbf{B}}_s^n) : \boldsymbol{\tau} = 0, \tag{3.1c}$$

$$\int_\Omega 3\varphi^n\psi = \int_\Omega 4\varphi^{n-1}\circ\mathsf{X}_{n-1}^{(1)}\,\psi - \varphi^{n-2}\circ\mathsf{X}_{n-2}^{(2)}\,\psi, \tag{3.1d}$$

for all $\boldsymbol{\nu} \in \mathbb{V}(0)$, $q \in \mathbb{Q}$, $\boldsymbol{\tau} \in \mathbb{W}$ and $\psi \in \mathbb{X}(0)$. Indeed, we first solve at each time step the fluid problem given the membrane position. The membrane is advected subsequently according to the state of the fluid. That is achieved through the transport of the level set function with the fluid velocity solution using the characteristics method. Given the membrane position (i.e. the level set solution after redistancing), we denote by $\boldsymbol{\chi} \equiv (\boldsymbol{u}, p, \hat{\mathbf{B}}_s)$ the global vector of unknowns, while $\mathscr{R}(\boldsymbol{\chi})$ represents the global residual vector corresponding to the system (3.1a)–(3.1c).

3.1.2. Consistent linearization and Newton–Raphson method

We first recall the standard Newton method. Assuming that a given functional $\mathcal{F}(\boldsymbol{\chi}(.))$ is Fréchet differentiable, let $D\mathcal{F}(\boldsymbol{\chi})[\delta\boldsymbol{\chi}]$ denote the Fréchet derivative of $\mathcal{F}$ at $\boldsymbol{\chi}$ along the direction $\delta\boldsymbol{\chi}$. Given the solutions at time t^{n-1}, we iteratively compute $\boldsymbol{\chi}_k^n$ at t^n, and the Newton–Raphson method reduces the system (3.1a)–(3.1c) into a sequence of linear sub-problems as follows:

$$\text{compute} \quad \boldsymbol{\chi}_{k+1}^n = \boldsymbol{\chi}_k^n + \delta\boldsymbol{\chi}_k^n \quad \text{such that} \quad \left\langle D\mathscr{R}\left(\boldsymbol{\chi}_k^n\right)\left[\delta\boldsymbol{\chi}_k^n\right], \boldsymbol{\xi}\right\rangle = -\left\langle \mathscr{R}\left(\boldsymbol{\chi}_k^n\right), \boldsymbol{\xi}\right\rangle, \quad \forall\boldsymbol{\xi}.$$

Thereafter, we introduce a generalization to the multidimensional case of the Kou's method [1]. It represents an improvement of the Weerakoon–Fernando method [48] and features an efficiency index (defined as the convergence order per residual or Jacobian evaluation) of $\sqrt[3]{3}$; that is higher than the efficiency index $\sqrt{2}$ of the standard Newton method, see [1]. The Kou's method features a third-order convergence behavior and requires two evaluations of the residual and one assembly and factorization of the Jacobian matrix. The utility of this method was proved in [1] in the case of zero-dimensional examples, where comparisons with other cubically convergent methods are also provided.

The choice of this method among other cubically convergent methods (in particular with respect to the cubically convergent method developed in [36]) is motivated by the fact that the computational cost of the Jacobian assembly and factorization is observed much higher than the computational cost of the residual's evaluation. Indeed, let us consider a mesh having 30′931 tetrahedra and set $\Delta t = 0.015$. We evaluate the computing times required for the Jacobian assembly and factorization and the residual's evaluation on a workstation with an Intel ® Core ™ i7-4790 (3.6 GHz) processor. The CPU time required for the residual's assembly 59.910 s is almost 4 times cheaper than the CPU time of the Jacobian assembly 61.161 s and factorization 155.303 s. Notice that the resolution of the resulting linear system is significantly cheaper 0.304 s. The Kou's method consists in computing the increment at each Newton iteration in two steps as follows:

(1) First, compute $\boldsymbol{\chi}_{k+0.5}^n = \boldsymbol{\chi}_k^n + \delta\boldsymbol{\chi}_k^n$ such that:

$$\left\langle D\mathscr{R}\left(\boldsymbol{\chi}_k^n\right)\left[\delta\boldsymbol{\chi}_k^n\right], \boldsymbol{\xi}\right\rangle = \left\langle \mathscr{R}\left(\boldsymbol{\chi}_k^n\right), \boldsymbol{\xi}\right\rangle, \quad \forall\boldsymbol{\xi}. \tag{3.2}$$

(2) Secondly, compte the solution $\boldsymbol{\chi}_{k+1}^n = \boldsymbol{\chi}_{k+0.5}^n + \delta\boldsymbol{\chi}_{k+0.5}^n$ such that:

$$\left\langle D\mathscr{R}\left(\boldsymbol{\chi}_k^n\right)\left[\delta\boldsymbol{\chi}_{k+0.5}^n\right], \boldsymbol{\xi}\right\rangle = -\left\langle \mathscr{R}\left(\boldsymbol{\chi}_{k+0.5}^n\right), \boldsymbol{\xi}\right\rangle, \quad \forall\boldsymbol{\xi}. \tag{3.3}$$

Thereafter, we drop the subscript n referring to the time step whenever it is contextually clear, and we rather denote k as a superscript. A second-order extrapolation of the solution at the previous time steps enables to set the starting values at each Newton loop. Since the Newton algorithm is sensible to the choice of the starting value, we sometimes use a fixed-point method to set the initial guess for the first nonlinear iteration. We iterate the method until reaching the convergence. The stopping criterion is based on the computation of the residual. We set the Newton tolerance to 10^{-8} in our numerical computations.

Let us focus on the computation of the directional derivatives of the invariants of the surface left Cauchy–Green deformation tensor $\mathbf{B}_s$ in the direction of $\delta\hat{\mathbf{B}}_s$. We have:

$$D\mathrm{tr}(\mathbf{B}_s)\left[\delta\hat{\mathbf{B}}_s\right] = \delta\hat{\mathbf{B}}_s : \boldsymbol{\pi}_\Gamma \quad \text{and} \quad D\mathrm{tr}\left(\mathbf{B}_s^2\right)\left[\delta\hat{\mathbf{B}}_s\right] = 2\delta\hat{\mathbf{B}}_s\boldsymbol{\pi}_\Gamma : \boldsymbol{\pi}_\Gamma\hat{\mathbf{B}}_s = 2\hat{\mathbf{B}}_s\boldsymbol{\pi}_\Gamma\delta\hat{\mathbf{B}}_s : \boldsymbol{\pi}_\Gamma.$$

We now proceed with the linearization of the strain invariant $\mathcal{J}_s$ which leads to:

$$\begin{aligned}
D\mathcal{J}_s\left[\delta\hat{\mathbf{B}}_s\right] &= \frac{1}{2\mathcal{J}_s}\left(\left(\hat{\mathbf{B}}_s : \boldsymbol{\pi}_\Gamma\right)\left(\delta\hat{\mathbf{B}}_s : \boldsymbol{\pi}_\Gamma\right) - \delta\hat{\mathbf{B}}_s\boldsymbol{\pi}_\Gamma : \boldsymbol{\pi}_\Gamma\hat{\mathbf{B}}_s\right) = \frac{1}{2\mathcal{J}_s}\left(\left(\hat{\mathbf{B}}_s : \boldsymbol{\pi}_\Gamma\right)\mathbf{I} - \hat{\mathbf{B}}_s\boldsymbol{\pi}_\Gamma\right)\delta\hat{\mathbf{B}}_s : \boldsymbol{\pi}_\Gamma \\
D\frac{1}{\mathcal{J}_s}\left[\delta\hat{\mathbf{B}}_s\right] &= \frac{-1}{\mathcal{J}_s^2}D\mathcal{J}_s\left[\delta\hat{\mathbf{B}}_s\right] = \frac{-1}{2\mathcal{J}_s^3}\left(\left(\hat{\mathbf{B}}_s : \boldsymbol{\pi}_\Gamma\right)\mathbf{I} - \hat{\mathbf{B}}_s\boldsymbol{\pi}_\Gamma\right)\delta\hat{\mathbf{B}}_s : \boldsymbol{\pi}_\Gamma, \\
D\frac{1}{\mathcal{J}_s^3}\left[\delta\hat{\mathbf{B}}_s\right] &= \frac{-3}{\mathcal{J}_s^4}D\mathcal{J}_s\left[\delta\hat{\mathbf{B}}_s\right] = \frac{-3}{2\mathcal{J}_s^5}\left(\left(\hat{\mathbf{B}}_s : \boldsymbol{\pi}_\Gamma\right)\boldsymbol{\pi}_\Gamma - \boldsymbol{\pi}_\Gamma\hat{\mathbf{B}}_s\boldsymbol{\pi}_\Gamma\right) : \delta\hat{\mathbf{B}}_s.
\end{aligned} \tag{3.4}$$

To make the finite element implementation straightforward and present the tangent problem in a compact manner, we consider the following weighted multi-linear forms. We use the semicolon symbol to separate between the unknown and the test function on one side and the weights on the other side. We introduce for all $\boldsymbol{u}$, $\boldsymbol{v} \in H^1(\Omega)^d$; $q \in \mathbb{Q}$; $\boldsymbol{\tau}, \mathbf{T} \in \mathbb{W}$; $w \in L^\infty(\Omega)$ and $\boldsymbol{P} \in L^\infty(\Omega)^{d\times d}$ the forms:

$$\begin{aligned}
&a(\boldsymbol{u}, \boldsymbol{v}; w) = \int_\Omega 2w\mathbf{D}(\boldsymbol{u}) : \mathbf{D}(\boldsymbol{v}); \quad b(\boldsymbol{u}, q; \boldsymbol{P}) = -\int_\Omega q\boldsymbol{P} : \boldsymbol{\nabla}\boldsymbol{u}; \quad c(\boldsymbol{u}, \boldsymbol{v}; w, \boldsymbol{w}) = \int_\Omega w\left(\left(\boldsymbol{u}\cdot\boldsymbol{\nabla}\right)\boldsymbol{w} + \left(\boldsymbol{w}\cdot\boldsymbol{\nabla}\right)\boldsymbol{u}\right).\boldsymbol{v}; \\
&d(\boldsymbol{\tau}, \boldsymbol{v}; w, \boldsymbol{P}) = \int_\Omega w\,\boldsymbol{\tau} : \left(\boldsymbol{P}\boldsymbol{\nabla}\boldsymbol{v}\boldsymbol{P}\right); \quad e(\mathbf{T}, \boldsymbol{v}; w, \boldsymbol{P}, \boldsymbol{\tau}) = \int_\Omega w\left(\left((\boldsymbol{\tau} : \boldsymbol{P})\mathbf{I} - \boldsymbol{\tau}\boldsymbol{P}\right)\mathbf{T} : \boldsymbol{P}\right)\left(\boldsymbol{\tau} : \boldsymbol{P}\boldsymbol{\nabla}\boldsymbol{v}\boldsymbol{P}\right); \\
&f(\mathbf{T}, \boldsymbol{v}; \boldsymbol{P}, \boldsymbol{\tau}) = \int_\Omega (\boldsymbol{\tau}\mathbf{T} : \boldsymbol{P})\left(\boldsymbol{\nabla}\boldsymbol{v} : \boldsymbol{P}\right); \quad m(\boldsymbol{u}, \boldsymbol{v}; w) = \int_\Omega w\,\boldsymbol{u}\cdot\boldsymbol{v}; \quad g(\mathbf{T}, \boldsymbol{\tau}) = \int_\Omega \mathbf{T} : \boldsymbol{\tau}; \\
&\alpha(\mathbf{T}, \boldsymbol{\tau}; \boldsymbol{u}) = \int_\Omega \left(\boldsymbol{u}\cdot\boldsymbol{\nabla}\mathbf{T} - \mathbf{sym}\left(\boldsymbol{\nabla}\boldsymbol{u}\mathbf{T}\right)\right) : \boldsymbol{\tau}; \quad \beta(\boldsymbol{u}, \boldsymbol{\tau}; \mathbf{T}) = \alpha(\mathbf{T}, \boldsymbol{\tau}; \boldsymbol{u}).
\end{aligned}$$

For ease of exposition, we only present the tangent system associated to the classical quadratically convergent Newton method. The semidiscrete approximation in time of the tangent problem reads:

$\mathscr{P}_{\varphi^{n-1}}(\delta\boldsymbol{\chi}^k)$: *Given* $\boldsymbol{\chi}^k$, *find the update* $\delta\boldsymbol{\chi}^k \equiv (\delta\boldsymbol{u}^k, \delta p^k, \delta\hat{\mathbf{B}}_s^k) \in \mathbb{V}(\boldsymbol{u}_b)\times\mathbb{Q}\times\mathbb{W}$ *such that*

$$\begin{aligned}
&\frac{3}{2\Delta t}m\Big(\delta\boldsymbol{u}^k,\boldsymbol{v};\rho_\varepsilon\big(\varphi^{n-1}\big)\Big)+c\Big(\delta\boldsymbol{u}^k,\boldsymbol{v};\rho_\varepsilon\big(\varphi^{n-1}\big),\boldsymbol{u}^k\Big)+a\Big(\delta\boldsymbol{u}^k,\boldsymbol{v};\mu_\varepsilon\big(\varphi^{n-1}\big)\Big)+b\Big(\boldsymbol{v},\delta p^k;\mathbf{I}\Big)\\
&+\frac{\mathsf{E}_s}{3}d\left(\delta\hat{\mathbf{B}}_s^k,\boldsymbol{v};\frac{1}{\mathcal{J}_s^k}|\nabla\varphi^{n-1}|\delta_\varepsilon(\varphi^{n-1}),\boldsymbol{\pi}_\Gamma\right)-\frac{\mathsf{E}_s}{3}e\left(\delta\hat{\mathbf{B}}_s^k,\boldsymbol{v};\frac{1}{2(\mathcal{J}_s^k)^3}|\nabla\varphi^{n-1}|\delta_\varepsilon(\varphi^{n-1}),\boldsymbol{\pi}_\Gamma,\hat{\mathbf{B}}_s^k\right)\\
&+\mathsf{E}_s f\left(\delta\hat{\mathbf{B}}_s^k,\boldsymbol{v};\boldsymbol{\pi}_\Gamma,\frac{|\nabla\varphi^{n-1}|}{2(\mathcal{J}_s^k)^5}\big((\hat{\mathbf{B}}_s^k:\boldsymbol{\pi}_\Gamma)\mathbf{I}-\hat{\mathbf{B}}_s^k\boldsymbol{\pi}_\Gamma\big)\delta_\varepsilon(\varphi^{n-1})\right)=-\Big\langle\mathscr{R}_{\boldsymbol{\chi}}\big(\boldsymbol{\chi}^k\big),\boldsymbol{v}\Big\rangle_{\mathbb{V}(\mathbf{0})',\mathbb{V}(\mathbf{0})},
\end{aligned}\tag{3.5a}$$

$$b(\delta\boldsymbol{u}^k,q;\mathbf{I})=-\langle\mathscr{R}_p(\boldsymbol{u}^k),q\rangle_{\mathbb{Q}',\mathbb{Q}},\tag{3.5b}$$

$$\frac{3}{2\Delta t}g\big(\delta\hat{\mathbf{B}}_s^k,\boldsymbol{\tau}\big)+\alpha\big(\delta\hat{\mathbf{B}}_s^k,\boldsymbol{\tau};\boldsymbol{u}^k\big)+\beta\big(\delta\boldsymbol{u}^k,\boldsymbol{\tau};\hat{\mathbf{B}}_s^k\big)=-\Big\langle\mathscr{R}_{\boldsymbol{u},\hat{\mathbf{B}}_s}(\boldsymbol{u}^k,\hat{\mathbf{B}}_s^k),\boldsymbol{\tau}\Big\rangle_{\mathbb{W}',\mathbb{W}},\tag{3.5c}$$

for all $\boldsymbol{v}\in\mathbb{V}(\mathbf{0})$, $q\in\mathbb{Q}$ and $\boldsymbol{\tau}\in\mathbb{W}$. The corresponding residuals are expressed as:

$$\begin{aligned}
\Big\langle\mathscr{R}_{\boldsymbol{\chi}}\big(\boldsymbol{\chi}^k\big),\boldsymbol{v}\Big\rangle=&\frac{1}{2\Delta t}m\Big(3\boldsymbol{u}^k-4\boldsymbol{u}^{n-1}+\boldsymbol{u}^{n-2},\boldsymbol{v};\rho_\varepsilon\big(\varphi^{n-1}\big)\Big)+\frac{1}{2}c\Big(\boldsymbol{u}^k,\boldsymbol{v};\rho_\varepsilon\big(\varphi^{n-1}\big),\boldsymbol{u}^k\Big)\\
&+a\Big(\boldsymbol{u}^k,\boldsymbol{v};\mu_\varepsilon\big(\varphi^{n-1}\big)\Big)+b\Big(\boldsymbol{v},p^k;\mathbf{I}\Big)+\frac{\mathsf{E}_s}{3}b\left(\boldsymbol{v},\frac{1}{(\mathcal{J}_s^k)^3}|\nabla\varphi^{n-1}|\delta_\varepsilon(\varphi^{n-1});\boldsymbol{\pi}_\Gamma^{n-1}\right)\\
&+\frac{\mathsf{E}_s}{3}d\left(\hat{\mathbf{B}}_s^k,\boldsymbol{v};\frac{1}{\mathcal{J}_s^k}|\nabla\varphi^{n-1}|\delta_\varepsilon(\varphi^{n-1}),\boldsymbol{\pi}_\Gamma^{n-1}\right)-m\Big(\boldsymbol{g},\boldsymbol{v};\rho_\varepsilon\big(\varphi^{n-1}\big)\Big),
\end{aligned}\tag{3.5d}$$

$$\langle\mathscr{R}_p(\boldsymbol{u}^k),q\rangle=b(\boldsymbol{u}^k,q;\mathbf{I}),\tag{3.5e}$$

$$\Big\langle\mathscr{R}_{\boldsymbol{u},\hat{\mathbf{B}}_s}(\boldsymbol{u}^k,\hat{\mathbf{B}}_s^k),\boldsymbol{\tau}\Big\rangle=\alpha\Big(\hat{\mathbf{B}}_s^k,\boldsymbol{\tau};\boldsymbol{u}^k\Big)+\frac{1}{2\Delta t}g\Big(3\hat{\mathbf{B}}_s^k-4\hat{\mathbf{B}}_s^{n-1}+\hat{\mathbf{B}}_s^{n-2},\boldsymbol{\tau}\Big).\tag{3.5f}$$

It is worth pointing that the tangent problem is derived from the time discrete system; hence it depends on the time advancing scheme chosen. Regarding the space discretization by finite elements, we adopt a Taylor-Hood finite element approximation for the discretization of $\boldsymbol{u}$ and p. Let $\boldsymbol{u}_{k,h}^n$ be an approximation of $\boldsymbol{u}_k^n$ at the Newton sub-iteration k at time t^n, and similarly for the other variables. We consider the following finite-dimensional spaces:

$$\begin{aligned}
&\mathbb{V}_h=\{\boldsymbol{u}_h\in\mathbb{V}(\boldsymbol{u}_b)\cap C^0(\overline{\Omega})^d:\boldsymbol{u}_{h|K}\in(\mathbb{P}_2)^d,\ \forall K\in\mathscr{T}_h\},\quad \mathbb{W}_h=\{\boldsymbol{\tau}_h\in\mathbb{W}:\boldsymbol{\tau}_h=\boldsymbol{\tau}_h^T,\boldsymbol{\tau}_{h,ij|K}\in\mathbb{P}_1,\forall K\in\mathscr{T}_h,1\leqslant i,j\leqslant d\},\\
&\mathbb{X}_h=\{\psi_h\in\mathbb{V}(\varphi_b)\cap C^0(\overline{\Omega}):\psi_{h|K}\in\mathbb{P}_2,\forall K\in\mathscr{T}_h\},\quad\text{and}\quad \mathbb{Q}_h=\{q_h\in C^0(\overline{\Omega}):q_{h|K}\in\mathbb{P}_1,\forall K\in\mathscr{T}_h\}.
\end{aligned}$$

The Galerkin scheme associated to the formulation $\mathscr{P}_{\varphi^{n-1}}$(3.5a)–(3.5c) consists in finding $\delta\boldsymbol{\chi}_{k,h}^n\in\mathbb{V}_h\times\mathbb{Q}_h\times\mathbb{X}_h$ at each Newton iteration. We subsequently search for $\delta\varphi_{k,h}^n\in\mathbb{X}_h$ as the solution of the discretized Eq. (3.1d).

Finally, Strategy I is strongly-coupled and consists in iterating at each time iteration between the subproblems (3.1a)–(3.1d) until reaching the convergence. This fixed-point strategy uses a convergence criterion based on the difference between two successive membrane positions, that is $\|\mathscr{H}_\varepsilon(\varphi^{l+1})-\mathscr{H}_\varepsilon(\varphi^l)\|_\infty<\epsilon_{Fp}$, where l represents the counter for the fixed-point iterations. This algorithm might additionally benefit from an adaptive relaxation parameter (e.g. generalized Aitken method [49]) but that was not investigated within this work. The overall coupling method is detailed in Algorithm 1.

3.2. Strategy II: fully implicit and monolithic strategy

In this section, we propose a monolithically-coupled solution scheme. In fact, the present model considers a massless membrane, which results in an infinite ratio of the added mass of the fluid to the structural mass and renders many standard partitioned iterative methods unstable [50]. That represents the main motivation to develop Strategy II.

Hereafter, all the linearities are treated implicitly and the entire problem is solved as a fully coupled system instead of solving separately the level set equation and the remaining subproblem. The time derivative terms are approximated by means of the second order backward differentiation formulas. In addition, we assume the signed distance assumption resulting from the resolution until convergence of the redistancing problem. That corresponds to the assumption $|\nabla\varphi|=1$ everywhere in Ω. If the redistancing problem is not iterated until reaching the convergence, the algorithm corresponds then to an inexact Newton strategy and the quadratic convergence of the Newton method can be deteriorated. Deep investigations of such inexact Newton strategies for the highly nonlinear problem of vesicles freely suspended in shear flow can be found in [51]. The time discrete fluid-membrane interaction problem reads:

Algorithm 1 (Strategy I): strongly-coupled segregated approach.

set initial conditions, Newton tolerance ϵ_N and fixed-point tolerance ϵ_{Fp}
let $n = 0$ and $\boldsymbol{\chi}^0$, φ^0 be known values
for $n = 1, \dots, n_{\max} = T/\Delta t$ **do**
  set $\varphi^{n,l=0} = \varphi^{n-1}$
  initialize $\boldsymbol{u}^n_{k=0} = \tilde{\boldsymbol{u}}^n$ and $\hat{\mathbf{B}}^n_{s,k=0} = \tilde{\hat{\mathbf{B}}}^n_s$ by second order extrapolated predictions
  for $l = 1, \dots$ **do**
    set initial increment $\delta\boldsymbol{\chi}^n_0 = \mathbf{0}$
    for $k = 1, \dots$ **do**
      assemble and factorize $D\mathscr{R}$ from $\mathscr{P}_{\varphi^{n,l-1}}(\delta\boldsymbol{\chi}^n_k)$ (3.5a)–(3.5c)
      evaluate the residual $\mathscr{R}\big(\boldsymbol{\chi}^n_k\big)$ from (3.5d)–(3.5f) using $\boldsymbol{\chi}^n_k$ and $\varphi^{n,l-1}$
      compute $\delta\boldsymbol{\chi}^n_k$ from (3.2)
      update $\boldsymbol{\chi}^n_{k+0.5} = \boldsymbol{\chi}^n_k + \delta\boldsymbol{\chi}^n_k$
      evaluate the residual $\mathscr{R}\big(\boldsymbol{\chi}^n_{k+0.5}\big)$ from (3.5d)–(3.5f) using $\boldsymbol{\chi}^n_{k+0.5}$ and $\varphi^{n,l-1}$
      compute $\delta\boldsymbol{\chi}^n_{k+0.5}$ from (3.3)
      update $\boldsymbol{\chi}^n_{k+1} = \boldsymbol{\chi}^n_{k+0.5} + \delta\boldsymbol{\chi}^n_{k+0.5}$
      if $\left\|\mathcal{R}_{\boldsymbol{\chi}}(\boldsymbol{\chi}^k)\right\|_{\mathbb{V}(\mathbf{0})'} < \epsilon_N$ **then**
        break
      end if
    end for
    update the solution $\boldsymbol{\chi}^{n,l} = \boldsymbol{\chi}^n_{k+1}$
    compute $\varphi^{n,l}$ from (2.3), advected with $\boldsymbol{u}^{n,l}$
    if $\left\|\mathscr{H}_\varepsilon(\varphi^{n,l}) - \mathscr{H}_\varepsilon(\varphi^{n,l-1})\right\|_\infty < \epsilon_{Fp}$ **then**
      break
    end if
  end for
  update the solution $\boldsymbol{\chi}^n = \boldsymbol{\chi}^{n,l}$ and $\varphi^n = \varphi^{n,l}$
end for

Strategy II: find $\boldsymbol{\chi}^n \equiv (\boldsymbol{u}^n, p^n, \hat{\mathbf{B}}^n_s, \varphi^n) \in \mathbb{V}(\boldsymbol{u}_b) \times \mathbb{Q} \times \mathbb{W} \times \mathbb{X}(\varphi_b)$ *such that*

$$\int_\Omega \rho_\varepsilon(\varphi^n)\left(\frac{3\boldsymbol{u}^n - 4\boldsymbol{u}^{n-1} + \boldsymbol{u}^{n-2}}{2\Delta t} + \boldsymbol{u}^n\cdot\nabla\boldsymbol{u}^n\right)\cdot\boldsymbol{v} + \int_\Omega 2\mu_\varepsilon(\varphi^n)\,\mathbf{D}(\boldsymbol{u}^n) : \mathbf{D}(\boldsymbol{v}) - \int_\Omega p^n\, div\,\boldsymbol{v}$$
$$-\frac{\mathsf{E}_s}{3}\int_\Omega \frac{1}{(\mathcal{J}^n_s)^3}\,\delta_\varepsilon(\varphi^n)\,\boldsymbol{\pi}^n_\Gamma : \nabla\boldsymbol{v} + \frac{\mathsf{E}_s}{3}\int_\Omega \frac{1}{\mathcal{J}^n_s}\,\delta_\varepsilon(\varphi^n)\,\hat{\mathbf{B}}^n_s : \boldsymbol{\pi}^n_\Gamma\nabla\boldsymbol{v}\boldsymbol{\pi}^n_\Gamma = \int_\Omega \rho_\varepsilon(\varphi^n)\,\boldsymbol{g}\cdot\boldsymbol{v}, \tag{3.6a}$$

$$\int_\Omega q\ \ div\,\boldsymbol{u}^n = 0, \tag{3.6b}$$

$$\int_\Omega \frac{3\hat{\mathbf{B}}^n_s - 4\hat{\mathbf{B}}^{n-1}_s + \hat{\mathbf{B}}^{n-2}_s}{2\Delta t} : \boldsymbol{\tau} + \int_\Omega \left(\boldsymbol{u}^n\cdot\nabla\hat{\mathbf{B}}^n_s\right) : \boldsymbol{\tau} - \int_\Omega \mathbf{sym}\left(\nabla\boldsymbol{u}^n\hat{\mathbf{B}}^n_s\right) : \boldsymbol{\tau} = 0, \tag{3.6c}$$

$$\int_\Omega \frac{3\varphi^n - 4\varphi^{n-1} + \varphi^{n-2}}{2\Delta t}\psi + \int_\Omega (\boldsymbol{u}^n\cdot\nabla\varphi^n)\psi = 0, \tag{3.6d}$$

defined for all $\boldsymbol{v} \in \mathbb{V}(0)$, $q \in \mathbb{Q}$, $\boldsymbol{\tau} \in \mathbb{W}$ and $\psi \in \mathbb{X}(0)$. Thereafter, we provide the directional derivatives of some useful quantities in the direction of a spatial increment $\delta\boldsymbol{\chi} \equiv (\delta\boldsymbol{u}, \delta p, \delta\hat{\mathbf{B}}_s, \delta\varphi)$. We have:

$$\begin{aligned} D\,div_s\,\boldsymbol{v}[\delta\varphi] &= D\boldsymbol{\pi}_\Gamma[\delta\varphi] : \nabla\boldsymbol{v} = -(\nabla\delta\varphi\cdot\nabla\boldsymbol{v})\cdot\nabla\varphi - (\nabla\varphi\cdot\nabla\boldsymbol{v})\cdot\nabla\delta\varphi,\\ D\mathcal{J}_s^{-3}[\delta\varphi] &= -3\mathcal{J}_s^{-4}\,D\mathcal{J}_s[\delta\varphi],\\ D\mathcal{J}_s[\delta\varphi] &= \frac{1}{4\mathcal{J}_s}D\Big(\big(\hat{\mathbf{B}}_s : \boldsymbol{\pi}_\Gamma\big)^2 - \big(\hat{\mathbf{B}}_s\boldsymbol{\pi}_\Gamma\big)^2 : \mathbf{I}\Big)[\delta\varphi] = \frac{1}{2\mathcal{J}_s}\Big(\mathrm{tr}(\mathbf{B}_s)\,\hat{\mathbf{B}}_s : D\boldsymbol{\pi}_\Gamma[\delta\varphi] - \hat{\mathbf{B}}_s\boldsymbol{\pi}_\Gamma\hat{\mathbf{B}}_s : D\boldsymbol{\pi}_\Gamma[\delta\varphi]\Big)\\ &= \frac{1}{2\mathcal{J}_s}\Big(\hat{\mathbf{B}}_s\boldsymbol{\pi}_\Gamma\hat{\mathbf{B}}_s\big(\nabla\varphi\otimes\nabla\delta\varphi + \nabla\delta\varphi\otimes\nabla\varphi\big) : \mathbf{I} - \big(\hat{\mathbf{B}}_s : \boldsymbol{\pi}_\Gamma\big)\big(\hat{\mathbf{B}}_s : \big(\nabla\varphi\otimes\nabla\delta\varphi + \nabla\delta\varphi\otimes\nabla\varphi\big)\big)\Big). \end{aligned} \tag{3.7}$$

Now, remembering (3.4) and (3.7), the full linearization of the surface stretching $\mathcal{J}_s$ reads:

$$\begin{aligned} D\mathcal{J}_s[\delta\boldsymbol{\chi}] &= D\mathcal{J}_s[\delta\hat{\mathbf{B}}_s] + D\mathcal{J}_s[\delta\varphi]\\ &= \frac{1}{2\mathcal{J}_s}\big((\hat{\mathbf{B}}_s : \boldsymbol{\pi}_\Gamma)\boldsymbol{\pi}_\Gamma - \boldsymbol{\pi}_\Gamma\hat{\mathbf{B}}_s\boldsymbol{\pi}_\Gamma\big) : \delta\hat{\mathbf{B}}_s + \frac{1}{2\mathcal{J}_s}(\hat{\mathbf{B}}_s\boldsymbol{\pi}_\Gamma\hat{\mathbf{B}}_s - (\hat{\mathbf{B}}_s : \boldsymbol{\pi}_\Gamma)\hat{\mathbf{B}}_s) : (\nabla\varphi\otimes\nabla\delta\varphi + \nabla\delta\varphi\otimes\nabla\varphi). \end{aligned}$$

Let us introduce the following weighted multi-linear forms:

$$i(\varphi,\psi;\boldsymbol{w})=\int_\Lambda \psi\boldsymbol{w}\cdot\nabla\varphi,\ \ h(\varphi,\psi;w)=\int_\Lambda w\varphi\psi,$$
$$j(\varphi,\boldsymbol{v};\boldsymbol{w})=\int_\Lambda \varphi\boldsymbol{v}\cdot\boldsymbol{w},\ \ r(\varphi,\boldsymbol{v};w,\boldsymbol{w})=a(\boldsymbol{v},\boldsymbol{w};\varphi w),$$
$$s(\varphi,\boldsymbol{v};\boldsymbol{P})=\int_\Lambda \varphi\boldsymbol{P}:\nabla\boldsymbol{v},$$
$$l(\varphi,\boldsymbol{v};w,\boldsymbol{w},\boldsymbol{\tau})=\int_\Lambda w\,\mathbf{sym}((\nabla\varphi\otimes\boldsymbol{w}+\boldsymbol{w}\otimes\nabla\varphi)\boldsymbol{\tau}):\nabla\boldsymbol{v},$$
$$\gamma(\varphi,\boldsymbol{v};\boldsymbol{w},\boldsymbol{P},\boldsymbol{\tau})=\int_\Lambda(\boldsymbol{P}:\nabla\boldsymbol{v})\ \boldsymbol{\tau}:(\nabla\varphi\otimes\boldsymbol{w}+\boldsymbol{w}\otimes\nabla\varphi),$$

defined for all $\boldsymbol{u},\boldsymbol{v}\in(H^1(\Lambda))^d; q\in L^2(\Lambda); w\in L^\infty(\Lambda); \boldsymbol{w}\in(L^\infty(\Lambda))^d; \varphi,\psi\in H^1(\Lambda)$and $\boldsymbol{P},\boldsymbol{\tau}\in(L^\infty(\Lambda))^{d\times d}$.

The residual corresponding to the advection equation of the level set function reads:

$$\left\langle\mathscr{R}_\varphi\left(\varphi^k,\boldsymbol{u}^k\right),\psi\right\rangle_{\mathbb{X}(0)',\mathbb{X}(0)}=h\left(\frac{3\varphi^k-4\varphi^{n-1}+\varphi^{n-2}}{2\Delta t},\psi;1\right)+i(\varphi^k,\psi;\boldsymbol{u}^k),\quad\forall\psi\in\mathbb{X}(0).$$

Collecting the expressions derived above, the exact tangent problem corresponding to Strategy II reads: find $\delta\boldsymbol{\chi}^k\equiv(\delta\boldsymbol{u}^k,\delta p^k,\delta\hat{\mathbf{B}}_s^k,\delta\varphi^k)\in\mathbb{V}(\boldsymbol{u}_b)\times\mathbb{Q}\times\mathbb{W}\times\mathbb{X}(\varphi^{n-1})$ such that

$$\begin{aligned}
&\frac{3}{2\Delta t}m\left(\delta\boldsymbol{u}^k,\boldsymbol{v};\rho_\varepsilon\left(\varphi^k\right)\right)+c\left(\delta\boldsymbol{u}^k,\boldsymbol{v};\rho_\varepsilon\left(\varphi^k\right),\boldsymbol{u}^k\right)+a\left(\delta\boldsymbol{u}^k,\boldsymbol{v};\mu_\varepsilon\left(\varphi^k\right)\right)+b\left(\boldsymbol{v},\delta p^k,\mathbf{I}\right)\\
&+(\rho_o-\rho_i)j\left(\delta\varphi^k,\boldsymbol{v};\delta_\varepsilon\left(\varphi^k\right)\left(\frac{3\boldsymbol{u}^k-4\boldsymbol{u}^{n-1}+\boldsymbol{u}^{n-2}}{2\Delta t}+\boldsymbol{u}^k\cdot\nabla\boldsymbol{u}^k\right)\right)+(\mu_o-\mu_i)r\left(\delta\varphi^k,\boldsymbol{v};\delta_\varepsilon\left(\varphi^k\right),\boldsymbol{u}^k\right)\\
&+\frac{\mathsf{E}_s}{3}d\left(\delta\hat{\mathbf{B}}_s^k,\boldsymbol{v};\frac{\delta_\varepsilon(\varphi^k)}{\mathcal{J}_s^k},\boldsymbol{\pi}_\Gamma^k\right)+\frac{\mathsf{E}_s}{2}f\left(\delta\hat{\mathbf{B}}_s^k,\boldsymbol{v};\boldsymbol{\pi}_\Gamma^k,\frac{\delta_\varepsilon(\varphi^k)}{(\mathcal{J}_s^k)^5}\left(\left(\hat{\mathbf{B}}_s^k:\boldsymbol{\pi}_\Gamma^k\right)\mathbf{I}-\hat{\mathbf{B}}_s^k\boldsymbol{\pi}_\Gamma^k\right)\right)\\
&-\frac{\mathsf{E}_s}{6}e\left(\delta\hat{\mathbf{B}}_s^k,\boldsymbol{v};\frac{\delta_\varepsilon(\varphi^k)}{(\mathcal{J}_s^k)^3},\boldsymbol{\pi}_\Gamma^k,\hat{\mathbf{B}}_s^k\right)+\frac{\mathsf{E}_s}{3}s\left(\delta\varphi^k,\boldsymbol{v};\frac{\delta'_\varepsilon(\varphi^k)}{\mathcal{J}_s^k}\boldsymbol{\pi}_\Gamma^k\hat{\mathbf{B}}_s^k\boldsymbol{\pi}_\Gamma^k-\frac{\delta'_\varepsilon(\varphi^k)}{(\mathcal{J}_s^k)^3}\boldsymbol{\pi}_\Gamma^k\right)\\
&+\frac{\mathsf{E}_s}{6}l\left(\delta\varphi^k,\boldsymbol{v};\frac{\delta_\varepsilon(\varphi^k)}{(\mathcal{J}_s^k)^3},\nabla\varphi^k,\mathbf{I}\right)-\frac{\mathsf{E}_s}{3}l\left(\delta\varphi^k,\boldsymbol{v};\frac{\delta_\varepsilon(\varphi^k)}{\mathcal{J}_s^k},\nabla\varphi^k,\hat{\mathbf{B}}_s^k\boldsymbol{\pi}_\Gamma^k\right)-(\rho_o-\rho_i)\,j\left(\delta\varphi^k,\boldsymbol{v};\delta_\varepsilon\left(\varphi^k\right)\boldsymbol{g}\right)\\
&+\frac{\mathsf{E}_s}{2}\gamma\left(\delta\varphi^k,\boldsymbol{v};\nabla\varphi^k,\frac{\delta_\varepsilon(\varphi^k)}{(\mathcal{J}_s^k)^5}\boldsymbol{\pi}_\Gamma^k-\frac{\delta_\varepsilon(\varphi^k)}{3(\mathcal{J}_s^k)^3}\boldsymbol{\pi}_\Gamma^k\hat{\mathbf{B}}_s^k\boldsymbol{\pi}_\Gamma^k,\hat{\mathbf{B}}_s^k\boldsymbol{\pi}_\Gamma^k\hat{\mathbf{B}}_s^k-\left(\hat{\mathbf{B}}_s^k:\boldsymbol{\pi}_\Gamma^k\right)\hat{\mathbf{B}}_s^k\right)=-\left\langle\mathscr{R}_{\boldsymbol{\chi}}\left(\boldsymbol{\chi}^k\right),\boldsymbol{v}\right\rangle_{\mathbb{V}(\mathbf{0})',\mathbb{V}(\mathbf{0})},
\end{aligned}\tag{3.8a}$$

$$b(\delta\boldsymbol{u}^k,q,\mathbf{I})=-\langle\mathscr{R}_p(\boldsymbol{u}^k),q\rangle_{\mathbb{Q}',\mathbb{Q}},\tag{3.8b}$$

$$\frac{3}{2\Delta t}g\left(\delta\hat{\mathbf{B}}_s^k,\boldsymbol{\tau}\right)+\alpha\left(\delta\hat{\mathbf{B}}_s^k,\boldsymbol{\tau};\boldsymbol{u}^k\right)+\beta\left(\delta\boldsymbol{u}^k,\boldsymbol{\tau};\hat{\mathbf{B}}_s^k\right)=-\left\langle\mathscr{R}_{\boldsymbol{u},\hat{\mathbf{B}}_s}(\boldsymbol{u}^k,\hat{\mathbf{B}}_s^k),\boldsymbol{\tau}\right\rangle_{\mathbb{W}',\mathbb{W}},\tag{3.8c}$$

$$h\left(\delta\varphi^k,\psi;\frac{3}{2\Delta t}\right)+i(\delta\varphi^k,\psi;\boldsymbol{u}^k)+j(\psi,\delta\boldsymbol{u}^k;\nabla\varphi^k)=-\langle\mathscr{R}_\varphi\left(\varphi^k,\boldsymbol{u}^k\right),\psi\rangle_{\mathbb{X}(0)',\mathbb{X}(0)},\tag{3.8d}$$

for arbitrary test functions $\boldsymbol{v}\in\mathbb{V}(\mathbf{0})$, $q\in\mathbb{Q}$, $\boldsymbol{\tau}\in\mathbb{W}$ and $\psi\in\mathbb{X}(0)$.

4. Numerical results

The present framework has been implemented within the C++ environment for scientific computing Rheolef [41]. Distributed-memory parallelism relies on MPI[1]. The library relies upon the Boost[2], Blas[3] and UMFPACK[4] packages for much of its functionalities. We have used MUMPS[5] for the factorization and as direct solver on distributed-memory architectures, together with the Scotch[6] ordering algorithm for minimizing the fill-in of the large and sparse matrices. Numerical results are displayed graphically using Paraview[7] and Gnuplot[8].

[1] Message Passing Interface - http://www.mpi-forum.org
[2] Boost libraries - http://www.boost.org
[3] Basic Linear Algebra Subprograms library - http://www.netlib.org/blas
[4] Umfpack routines - http://www.cise.ufl.edu/research/sparse/umfpack/
[5] MUMPS - http://graal.ens-lyon.fr/MUMPS
[6] Scotch - http://www.labri.fr/perso/pelegrin/scotch
[7] Paraview - http://www.paraview.org
[8] Gnuplot - http://www.gnuplot.info

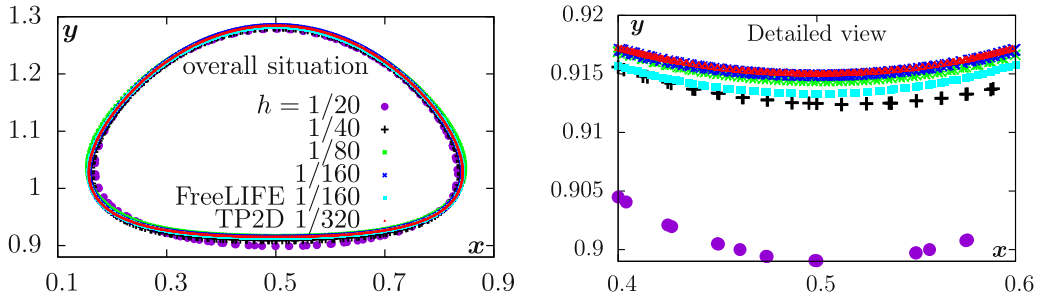


Fig. 4. Example 1: (left) comparison of the shapes at final time $t = 3$ with respect to the reference solutions in [53]. (right) Detailed view in the zone of maximal disparities at $t = 3$ (using the same colorbar). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article).

For numerical validations, this framework is investigated through different sets of membrane properties. In the first example, we restrict our attention to the particular case where the problem is reduced to the modeling of multiphase flows under the capillary effect. The second example concerns the pressurized flexible membrane where the mechanics is restricted to the membrane stretching [52]. In the third example, we test the method for a stiff problem involving a more complicated configuration such as the flower shape. Finally, we consider the full elasticity model in the three-dimensional case, and we provide insights on the computational timings required for both strategies.

4.1. Example 1: multiphase flow with surface tension effect

Although the main focus of this work is on the fluid-structure interaction problems involving thin elastic structures, we show here that the presented method can be easily adapted to solve the capillarity problem [53]. In such a case, the stored energy is linear with respect to the membrane stretching, that is $\mathcal{W}'(\mathcal{J}_s) = \gamma$ is constant. Here the surface tension coefficient γ is assumed isotropic and isothermal. Accordingly, the stress tensor is expressed as $\boldsymbol{\sigma}_s = \gamma \boldsymbol{\pi}_\Gamma$, and the corresponding capillary force $\mathcal{F}_\Gamma = -\gamma H \mathbf{n}$ follows from (2.2). To assess the performances and convergence properties of the strongly-coupled segregated approach, we solve the capillarity problem using Strategy I and we perform comparisons with some available results in the published literature.

We consider the benchmark of the rising bubble dynamics introduced in [53], and we proceed with the validation of the multiphase flow solver. The benchmark focuses on the dynamics of a circular bubble of radius $r_0 = 0.25$ initially centered at $\boldsymbol{x} = (0.5, 0.5)$ in the computational domain $[0,\ 1] \times [0,\ 2]$. We track the dynamics of the bubble for $t \in (0, T) = (0, 3)$. The surrounding fluid is heavier than the inner fluid such that $\rho_o = 1000$ and $\rho_i = 100$. The dynamics viscosities are set to $\mu_o = 10$ and $\mu_i = 1$, while the surface tension is $\gamma = 24.5$. No-slip boundary conditions are prescribed on the top and bottom boundaries, while free slip conditions $\boldsymbol{u} \cdot \boldsymbol{\nu} = 0$ and $\boldsymbol{t} \cdot \mathbf{D}(\boldsymbol{u}) \cdot \boldsymbol{\nu} = 0$ are imposed on the remaining boundaries, where $\boldsymbol{t}$ is a unit tangent vector on $\partial\Omega$.

For a quantitative study, we consider some benchmark quantities following the reference test [53]. Let u_y be the y-component of the velocity vector. The degree of circularity $\phi(t)$ measures the circumference of a circle having the same enclosed area as the interface. In addition, we introduce the rise velocity $V_{\mathbf{c}}(t)$ and the center of mass $Y_{\mathbf{c}}(t)$ given by:

$$\phi(t) = \frac{2\sqrt{\pi\,|\Omega_i|(t)}}{|\Gamma(t)|}, \quad Y_{\mathbf{c}}(t) = \frac{1}{|\Omega_i|(t)} \int_{\Omega_i(t)} y(t) d\mathbf{x} \quad \text{and} \quad V_{\mathbf{c}}(t) = \frac{1}{|\Omega_i|(t)} \int_{\Omega_i(t)} u_y(t) d\mathbf{x}.$$

We also compute the minimum circularity $\phi_{\min}$, the corresponding incidence time $t|_{\phi=\phi_{\min}}$, the maximal velocity $V_{\mathbf{c},\,\max}$ and the corresponding time incidence $t|_{V_{\mathbf{c}}=V_{\mathbf{c},\max}}$. The enclosed area $|\Omega_i|(t) = \pi/16$ should be preserved. We proceed with a comparative study with the reference solutions obtained in [53] and referred to as "TP2D", "FreeLIVE" and "MooNMD" (Finite Element method). We also compare to [54] (Volume-of-Fluid method), [55] (Volume-of-Fluid method), [56] (adaptive Finite Element method) and [57] (Finite Element method).

We perform computations on regular meshes. The mesh resolutions are $h \in \{1/20, 1/40, 1/80, 1/160\}$ and the corresponding time step sizes are $\Delta t \in \{3.5 \times 10^{-2}, 2 \times 10^{-2}, 10^{-2}, 2 \times 10^{-3}\}$, respectively. In Fig. 4, we compare the shapes obtained at the final time $t = 3$ using different meshes. The zoom view in the zone of maximum disparities show a good congruence between the shape obtained with the finest mesh and the reference solutions. In Fig. 5, some snapshots show the pressure discontinuous across the interface Γ. The temporal evolution of $y_{\mathbf{c}}$ and $V_{\mathbf{c}}$ depict that no significant differences can be observed at all, see Fig. 6. Only the zoom views show small deviations that are minimized for the finest resolutions. In Table 1 we compare the computed benchmark quantities to those obtained in the reference papers. Overall, a good agreement of the computations on the finest meshes is observed.

The spatial accuracy of the numerical approximation of a given quantity ζ is studied by computing normalized errors on successively refined meshes with respect to a reference solution $\tilde{\zeta}$ obtained with the finest mesh size. The time step is small

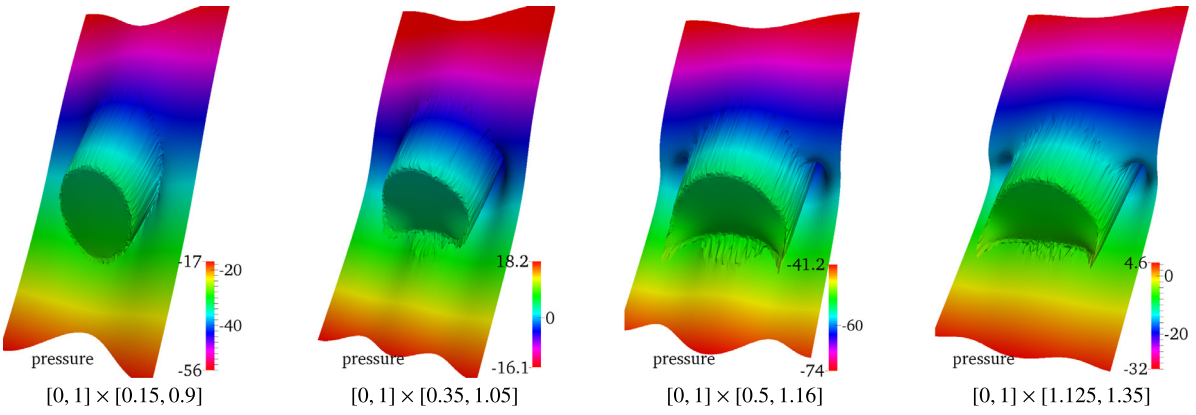


Fig. 5. Example 1: snapshots showing the pressure jump across the membrane at times $t = 0.461, 1.385, 2.077, 2.977$, respectively. A mesh size $h = 1/100$ is used.

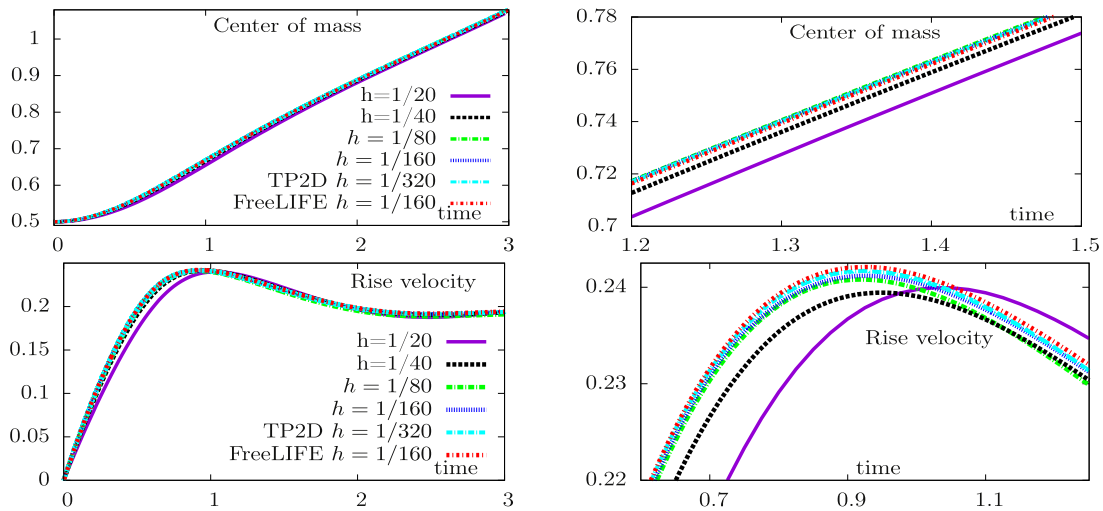


Fig. 6. Example 1: comparisons of the temporal evolution of the center of mass (Top) and rise velocity curves (Bottom) with respect to the reference solutions [53]. (Left) The complete simulation period. (Right) Detailed view in zones of maximum discrepancies using the same colorbar. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article).

Table 1
Example 1: comparisons with benchmark quantities and published results: minimum circularity and maximum rise velocity, with corresponding incidence times, and the final position of the center of mass.

	Computations				Available results				
	$1/h = 20$	40	80	**160**	benchmark [53]	Ref. [54]	Ref. [55]	Ref. [56]	Ref. [57]
$\phi^t_{\min}$	0.9161	0.9060	0.9021	**0.90107**	0.9012 ± 0.0001	0.8876	0.9044	0.9013	0.9001
$t\|_{\phi=\phi_{\min}}$	2.0000	1.9285	1.9142	**1.90430**	1.8895 ± 0.0145	1.8915	1.9625	1.8835	1.9000
$V_{\mathbf{c},\max}$	0.2399	0.2412	0.2415	**0.24163**	0.2419 ± 0.0002	0.2457	0.2348	0.2419	0.2412
$t\|_{V_{\mathbf{c}}=V_{\mathbf{c},\max}}$	1.0250	0.9571	0.9398	**0.93146**	0.9263 ± 0.0050	0.9235	0.9516	0.9290	0.9248
$Y_{\mathbf{c}}(t = 3)$	1.0699	1.0779	1.0812	**1.08177**	1.0808 ± 0.0009	1.0679	1.0696	1.0809	1.0815

enough to avoid significantly influencing the overall accuracy. We introduce the following errors and rate of convergence:

$$\|e_\zeta\|_1 = \frac{\sum_{t=1}^{N} |\tilde{\zeta} - \zeta|}{\sum_{t=1}^{N} |\tilde{\zeta}|}, \quad \|e_\zeta\|_2^2 = \frac{\sum_{t=1}^{N} |\tilde{\zeta} - \zeta|^2}{\sum_{t=1}^{N} |\tilde{\zeta}|^2}, \quad \|e_\zeta\|_\infty = \frac{\max_t |\tilde{\zeta} - \zeta|}{\max_t |\tilde{\zeta}|}, \quad \text{and } r_{h,i} = \frac{\log(\|e\|_i/\|\hat{e}\|_i)}{\log(h/\hat{h})}, \tag{4.1}$$

with $i \in \{1, 2, \infty\}$, while e and $\hat{e}$ stand for the errors obtained with two consecutive meshes of sizes h and $\hat{h}$, respectively. We consider the standard linear interpolation to account for the differences in the time step sizes. We perform simulations for the time interval (0,1). Table 2 shows the error history and the corresponding orders of convergence. Computations

Table 2
Example 1: convergence history of some benchmark quantities with respect to the spatial resolution.

	$1/h$	$\|e\|_1$	$r_{h,1}$	$\|e\|_2$	$r_{h,2}$	$\|e\|_\infty$	$r_{h,\infty}$
	20	6.024E−3		5.241E−3		4.408E−3	
ϕ	40	9.300E−4	2.695	9.455E−4	2.471	1.249E−4	1.820
	80	1.336E−4	2.799	1.616E−4	2.548	3.210E−4	1.960
	20	4.662E−3		5.812E−3		1.106E−2	
Y_c	40	1.686E−3	1.467	2.035E−3	1.514	3.714E−3	1.574
	80	6.006E−4	1.489	7.321E−4	1.475	1.362E−3	1.447
	20	1.586E−1		1.545E−1		1.399E−1	
V_c	40	3.977E−2	1.995	3.866E−2	1.998	3.462E−2	2.015
	80	9.542E−3	2.059	9.411E−3	2.039	9.211E−3	1.910
	20	8.027E−3		8.027E−3		8.037E−3	
$\|\Omega_i\|$	40	1.509E−3	2.411	1.509E−3	2.411	1.516E−3	2.406
	80	2.364E−4	2.675	2.364E−4	2.675	2.394E−4	2.663

reveal that the circularity approaches a convergence order of 2.75 and 2.5 in the l_1 and l_2 norms, respectively, whereas the convergence order is about 2 in the l_∞ norm. The convergence orders of the center of mass decrease to 1.5 in all norms. The rise velocity has a convergence order of about 2 in all norms. Finally, the area of the enclosed domain approaches a convergence order of about 2.75 in all norms.

4.2. Example 2: relaxation of stretched elastic membrane

This numerical example concerns the relaxation of a pressurized flexible membrane in the two-dimensional case under the effect of the elastic force. Here the elastic properties are restricted to the membrane stretching [52], while the stress-strain relationship will be precised afterwards. We consider the set-up described in [58,59] of the ellipsoidal membrane having initially the semi-major and semi-minor axes $a = 0.75$ and $b = 0.5$, respectively, and immersed in a fluid having $\rho_i = \rho_o = 1$. We consider constant viscosities in the inner and outer domains $\mu \equiv \mu_i = \mu_o$. The membrane is centered in a square computational domain and we prescribe homogeneous Dirichlet boundary conditions for the velocity (i.e. $\boldsymbol{u}_b = \mathbf{0}$).

The resting unstretched membrane is circular and has the radius 0.5. The restoring force brings the stretched ellipse back to an equilibrium circular steady state (but stretched) with the same enclosed area and having a radius $\sqrt{ab} \approx 0.61237$ (enclosed area preserved), in which the force vanishes. Accordingly, the membrane stretching at the initial time is $\mathcal{J}_s(0) \approx 1.2625316$ and corresponds to the surface stress tensor $\hat{\mathbf{B}}_s(0) = \mathcal{J}_s^2(0)\boldsymbol{\pi}_\Gamma$.

4.2.1. Convergence properties of the Newton algorithm

The membrane is centered in $[-1.5, 1.5]^2$, and we consider a linear stress-strain relationship such that the membrane force is given by: $\mathcal{F}_\Gamma = \mathbf{div}_s(\mathcal{W}'(\mathcal{J}_s)\boldsymbol{\pi}_\Gamma) = \boldsymbol{\nabla}_s(T_0(\mathcal{J}_s - 1)\boldsymbol{\pi}_\Gamma)$, where T_0 is the membrane tension coefficient set to 10. In Fig. 7, the temporal evolutions of the semi-axes and circularity for several values of the viscosity μ show higher frequencies of the membrane oscillations when considering lower viscosities until the convergence is reached. The results show a good agreement with [58,59] and are qualitatively comparable to those obtained in [52]. As pointed out above, the case of constant $\mathcal{J}_s(t), \forall t$ corresponds to the case of a bubble under the surface tension T_0 effect. Fig. 7 shows that the bubble's relaxation is slower than the relaxation of the elastic membrane. We proceed with a numerical investigation of the convergence properties of the Newton methods. We set the physical parameters to $\mu = 0.2$ and $T_0 = 1$. At every Newton sub-iteration, we compute the rate of convergence:

$$r(k) = \log\left(\frac{\left|\mathscr{R}\left(\boldsymbol{\chi}_{k,h}^n\right)\right|_{\mathbb{V}_h'(\mathbf{0})}}{\left|\mathscr{R}\left(\boldsymbol{\chi}_{k-1,h}^n\right)\right|_{\mathbb{V}_h'(\mathbf{0})}}\right) \Big/ \log\left(\frac{\left|\mathscr{R}\left(\boldsymbol{\chi}_{k-1,h}^n\right)\right|_{\mathbb{V}_h'(\mathbf{0})}}{\left|\mathscr{R}\left(\boldsymbol{\chi}_{k-2,h}^n\right)\right|_{\mathbb{V}_h'(\mathbf{0})}}\right), \quad \text{with } k > 2. \tag{4.2}$$

In Fig. 8 we report the rates of convergence $r(k)$ for several values of the time step size Δt. Remark that, by increasing Δt, the starting values for the Newton loop become far from the expected solution. Fig. 8 clearly depicts that the third-order convergence of the present method was obtained, whereas we observe a second-order convergence when using the standard Newton approach. For the largest values of Δt, we observe that the residuals curves have two parts: a plateau holds during the first iterations in which the residuals decrease slowly up to a threshold value, beyond it the residual converges with the expected quadratic or cubic behavior. Beyond a critical time step Δt, a complete breakdown of the algorithm is observable.

Hereafter, we focus on the computational cost of the present strategy and we perform comparisons with the classical Newton method. For a quantitative comparison, we evaluate the CPU times for serial simulations which enables to measure, together with the errors evaluation, the required computational effort to establish a certain accuracy.

We perform computations with the same regular mesh having 4′298 elements. In Table 3, we provide the timings of the computations for both strategies for several values of the time step size. We can observe that the quadratically convergent Newton approach behaves cheaper than the present method when considering very small time steps, almost for $\Delta t \leqslant 10^{-2}$. Indeed, the expected solution is relatively close to the starting value $\boldsymbol{\chi}^{k=0}$ and both methods converge in less than two iterations. Compared to the standard Newton method, the present approach requires an additional evaluation of the residual in

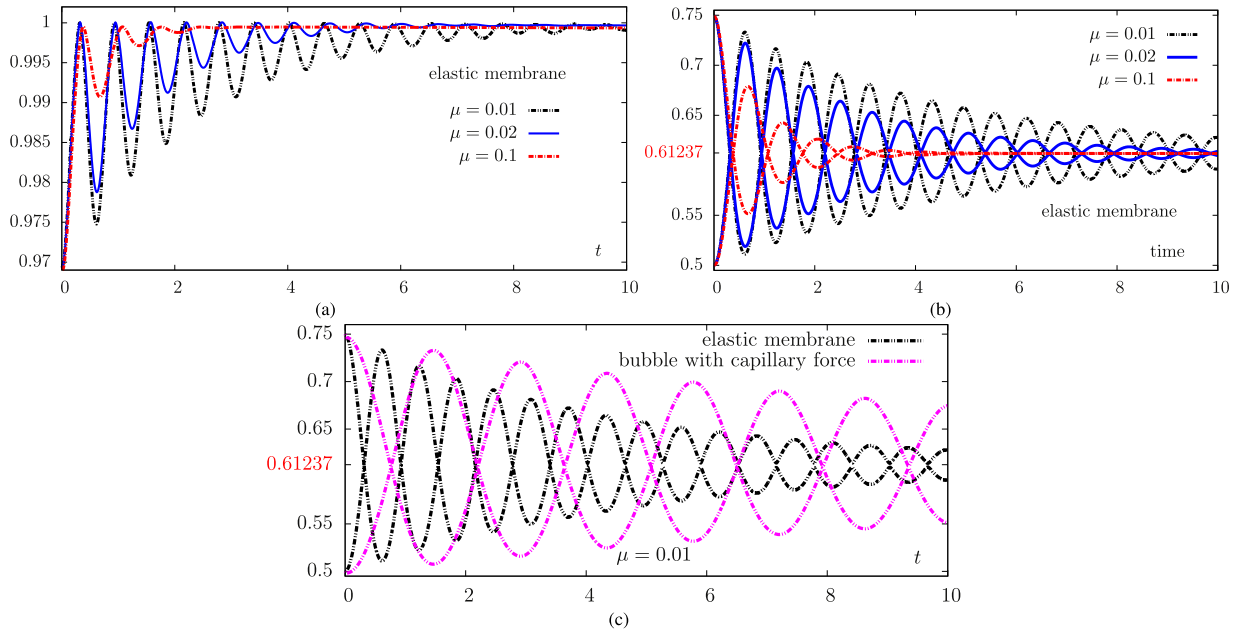
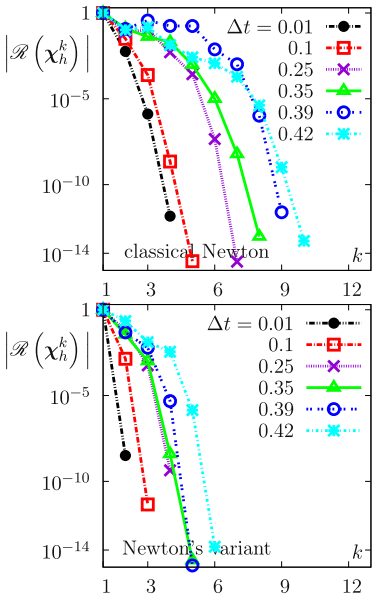


Fig. 7. Example 2: temporal evolution of the circularity ϕ (a) and the ellipse axes (b) for several values of the viscosity μ. (c) Comparison with respect to a relaxing bubble under the surface tension effect for the same parameters $T_0 = 10$ and $\mu = 0.01$.



Standard Newton method

k	$\Delta t = 0.01$	0.1	0.25	0.35	0.42
3	1.865	1.404	– –	0.519	0.163
4	1.793	2.383	0.971	0.654	6.464
5		1.254	2.964	4.310	0.673
6			1.888	1.478	0.496
7				1.636	2.244
8				1.489	2.125
9					2.162
10					1.192

Kou's method

k	$\Delta t = 0.01$	0.1	0.25	0.35	0.42
3	–	2.971	1.655	1.072	0.678
4			3.027	3.490	3.467
5				1.143	3.085

Fig. 8. Example 2: convergence properties of the cubically convergent Newton variant and comparison with the classical Newton method. (Left) Plots of the residuals' convergence curves. (Right) Corresponding rates of convergence.

Table 3
Example 2: comparison of the computational cost between the present Newton variant and the standard Newton method for several Δt.

	CPU time	
Δt	Kou's method	Standard Newton method
0.001	24.0345	15.7549
0.01	25.9083	23.0395
0.2	51.6319	57.8756
0.45	53.3225	74.5114

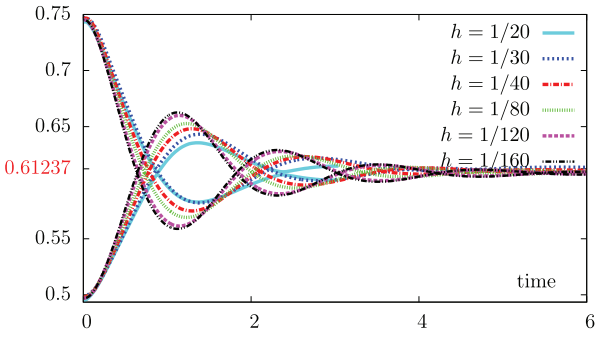


Fig. 9. Example 2: time evolution of the membrane major and minor axes for several spatial resolutions.

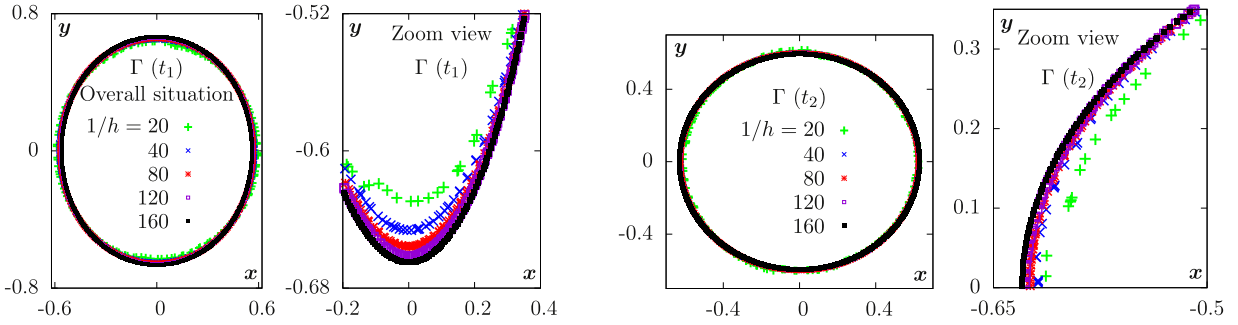


Fig. 10. Example 2: comparisons between the membrane shapes at $t_1 = 1.5$ and $t_2 = 2.33$ for different mesh sizes. The same color code is used. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article).

each Newton sub-iteration and leads to a higher computational cost. However, the present method becomes more beneficial when increasing the time step sizes, thanks to the third-order convergence behavior.

4.2.2. Accuracy tests

To assess the properties of the formulation, we perform a numerical study of the spatial convergence. The pressurized elastic membrane is centered in the computational domain $[-1.25, 1.25]^2$, while the stress-strain relationship is given by the relation $\mathcal{W}'(\mathcal{J}_s) = T_0(\mathcal{J}_s^2 - 1)$ with $T_0 = 2$. That corresponds to the elastic force $\mathcal{F}_\Gamma = div_s\,(T_0(\mathcal{J}_s^2 - 1)\boldsymbol{\pi}_\Gamma)$. We consider constant viscosities $\mu_i = \mu_o = 0.05$ and densities $\rho_i = \rho_o = 1$.

Fig. 12 shows some snapshots of the numerical solutions at different times during the oscillation period obtained with a mesh size $h = 1/160$. In particular, the pressure discontinuities across the membrane are clearly observed.

Let us consider a refined reference solution, referred to by the tilde symbol, obtained with the finest mesh size $h = 1/160$ and smallest time step size. Let $\|.\|_{0,\,\Omega}$ and $\|.\|_{1,\,\Omega}$ design the L^2 norm and the H^1 semi-norm, respectively. We numerically investigate the spatial accuracy of the proposed numerical scheme by computing the following errors on successively refined meshes with respect to the reference solution:

$$e_h(\boldsymbol{u}) = \frac{\|\boldsymbol{u}_h(.,T) - \tilde{\boldsymbol{u}}(.,T)\|_{1,\Omega}}{\|\tilde{\boldsymbol{u}}(.,T)\|_{1,\Omega}}, \quad e_h(p) = \frac{\|p_h(.,T) - \tilde{p}(.,T)\|_{1,\Omega}}{\|\tilde{p}(.,T)\|_{0,\Omega}}, \quad e_h(|\Omega_i|) = \frac{|\Omega_{i,h}| - \pi ab}{\pi ab},$$

$$e_h(\hat{\mathbf{B}}_s) = \frac{\|\mathrm{tr}\hat{\mathbf{B}}_{s,h}(.,T) - \mathrm{tr}\tilde{\hat{\mathbf{B}}}_s(.,T)\|_{0,\Gamma}}{\|\mathrm{tr}\tilde{\hat{\mathbf{B}}}_s(.,T)\|_{0,\Gamma}}, \quad e_h(\varphi) = \frac{\|\varphi_h(T,.) - \tilde{\varphi}(T,.)\|_{0,\Omega}}{\|\tilde{\varphi}(T,.)\|_{0,\Omega}} \quad \text{and} \quad e_h(|\Gamma|) = \frac{|\Gamma_h| - |\tilde{\Gamma}|}{|\tilde{\Gamma}|}.$$

The membrane shapes at particular times $t_1 = 1.35$ and $t_2 = 3$ are shown for several spatial resolutions in Fig. 10. The detailed views at zones of maximum discrepancies depict a good agreement. The time evolution of the major and minor axes for successively refined meshes are provided in Fig. 9. The area enclosed by the membrane should be preserved during the relaxation process. In Fig. 11, we provide the time evolution of the enclosed area, showing minor differences for the finest meshes with respect to the theoretical value. Notice that $|\Omega_{i,\,h}(t)|$ is not exactly preserved but it should only be satisfied as $h \to 0$. The evolution of the errors $e_h(|\Omega_i|)$ and $e_h(|\Gamma|)$ against the spatial resolution shows a second order accuracy. That is similar to the convergence behavior obtained with the Immersed Interface Method, while first order convergence was obtained with the standard IB method, see e.g. [52].

The spatial errors and the corresponding convergence rates for the numerical solutions with respect to the reference solution are depicted in Table 4, as function of the mesh size h. Even if the rigorous theoretical estimates are not available for the present problem formulation, the analogous estimates for elasticity and fluid problems suggest that the obtained

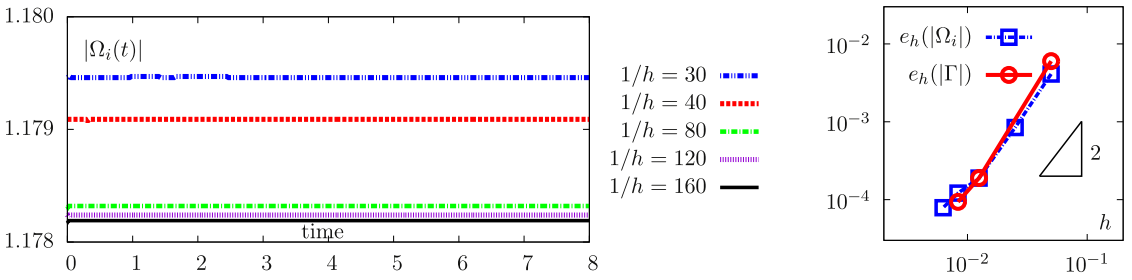


Fig. 11. Example 2: (Left) Time evolution of the enclosed area for several spatial resolutions. (Right) Spatial convergence curves of $|\Omega_i|$ and $|\Gamma|$.

Table 4
Example 2: spatial error history associated to the finite element discretization of the problem.

h	$e_h(\boldsymbol{u})$	$r_h(\boldsymbol{u})$	$e_h(p)$	$r_h(p)$	$e_h(\varphi)$	$r_h(\varphi)$	$e_h(\hat{\mathbf{B}}_s)$	$r_h(\hat{\mathbf{B}}_s)$
0.05	1.127800		0.869478		0.0798112		0.0844169	
0.025	0.883618	0.35202	0.581615	0.580084	0.0302727	1.39857	0.0535434	0.656821
0.0125	0.571068	0.62976	0.309685	0.909262	0.0135248	1.16241	0.0328551	0.704592
0.0083	0.442466	0.62926	0.217991	0.865926	0.0090735	0.98447	0.0247059	0.703055

Table 5
Example 2: relaxing stretched membrane in an incompressible flow. Relative error norms in the membranes axis and circularity and the corresponding orders of convergence.

mesh size	$\|e_\varpi\|_1$	$r_{h,1}(\varpi)$	$\|e_\varpi\|_2$	$r_{h,2}(\varpi)$	$\|e_\varpi\|_\infty$	$r_{h,\infty}(\varpi)$
1/40	3.8468E−2	–	4.7852E−2	–	8.4052E−2	–
1/80	1.0156E−2	1.9213	1.1948E−2	2.0018	1.9296E−2	2.123
1/120	4.0072E−3	2.2936	4.6708E−3	2.3165	7.4224E−3	2.3562
mesh size	$\|e_\phi\|_1$	$r_{h,1}(\phi)$	$\|e_\phi\|_2$	$r_{h,2}(\phi)$	$\|e_\phi\|_\infty$	$r_{h,\infty}(\phi)$
1/40	2.8730E−3	–	3.4529E−3	–	6.5534E−3	–
1/80	7.3916E−4	1.9586	9.7278E−4	1.8276	1.9719E−3	1.7327
1/120	3.0410E−4	2.1905	4.0989E−4	2.1315	8.6894E−4	2.021

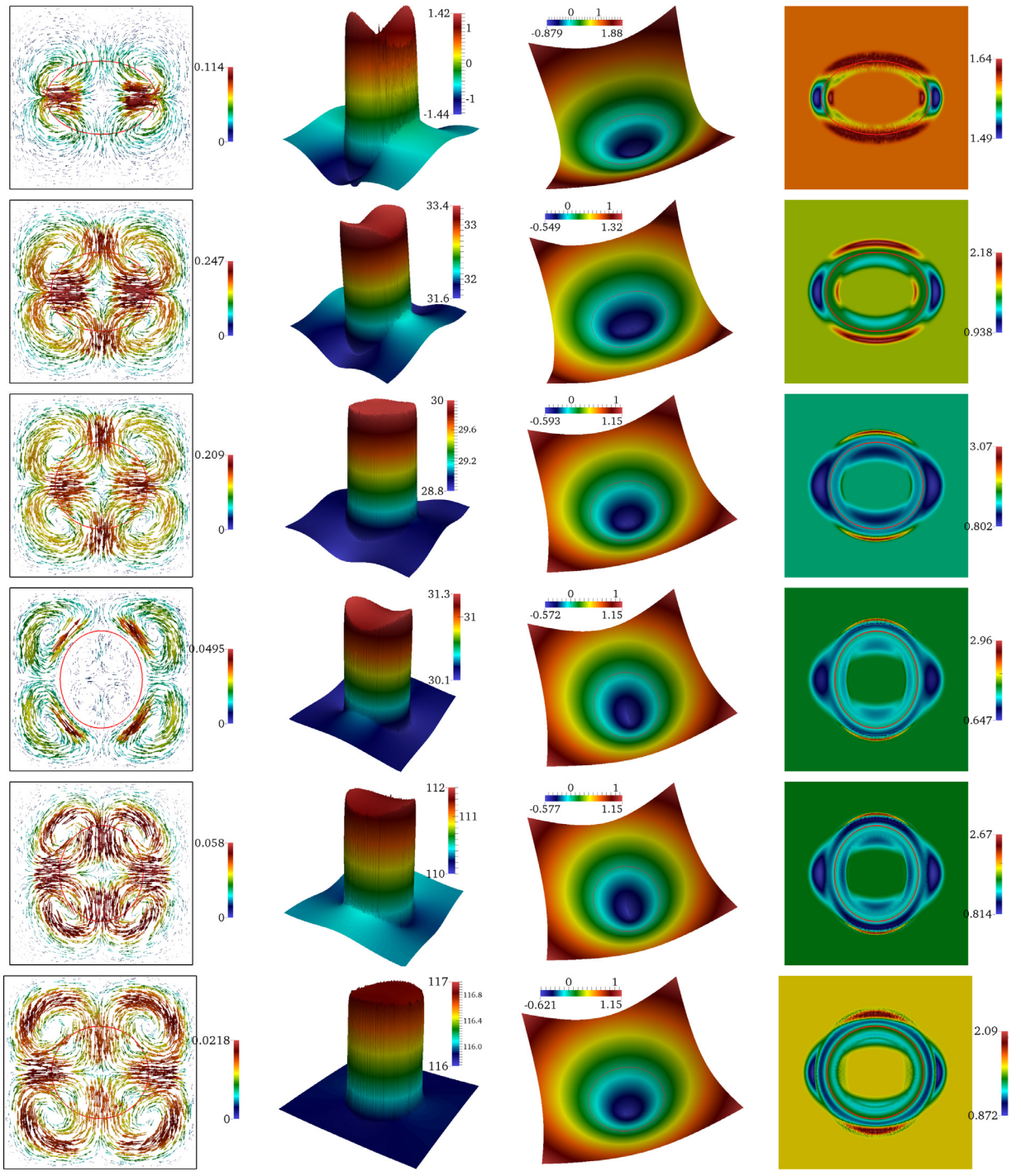
rates for the velocity and pressure are not optimal. The elastic force is singular and localized on the membrane, and the asymptotic convergence needs to consider $h << \varepsilon$. Most importantly, the lack of accuracy is due to the regularization of the elastic force on the membrane which results from the smoothing of the Dirac delta function. The spreading of the force generates larger forces in a small surrounding of the membrane. While the pressure field should be discontinuous across the membrane, the regularization of the force and the inability of the underlying finite element space to reproduce discontinuities inside the mesh elements reduce the numerical accuracy of the method, see e.g. the results and discussions in [20,52,60,61] for the IB-type method. To achieve the optimal rates of convergence when the solution is regular enough, one needs to retrieve the sharp representation of the membrane and impose the proper stress discontinuities across the membrane, see e.g. [62].

Very briefly, we present some possible techniques to treat this problem. The first technique require appropriate refinement and reconstruction of conforming meshes to accurately capture the membrane [62]. This approach can lead to a high computational burden for time-dependent problems. A variant of this approach is the Conformal Decomposition Finite Element Method (closely related to XFEM) that consists in cutting the elements which are intersected by the zero-level set curve or surface [63]. Several numerical issues can arise due to the generation of nearly degenerate elements, which may result in the ill-conditioning of the resulting linear system. Other methods consist in adapting the finite element approximation space instead of adapting the mesh. These *enriched approaches* consist in modifying or locally adding suitable basis functions so that they allow capturing the solution discontinuities across the membrane, see e.g. [64] for a review. These available techniques need to be investigated and adapted to the present framework to achieve the optimal convergence properties. That remains beyond the scope of this work and is currently investigated in an ongoing work.

Thereafter, we follow the numerical study in Example 1 and we compute the normalized errors on successively refined meshes for the membrane's axis ϖ and circularity ϕ. We consider the time horizon $t \in (0, T = 2.8)$ that corresponds to the first period of oscillations. In Table 5 we evaluate the errors given by (4.1) and we provide the corresponding rates of convergence. Numerical results suggest that the membrane's axis and circularity approach almost a convergence order of 2 in the l_1 and l_2 and l_∞ norms.

4.2.3. Influence of the physical parameters on the convergence of Strategy I and Strategy II

In this paragraph we investigate numerically the influence of the physical parameters on the convergence properties of the strongly-coupled partitioned and monolithic strategies. Let us consider the same setup described in 4.2.2. First, we focus



(a) $\boldsymbol{u}_h$ (b) p_h (c) φ_h (a) $\mathcal{J}^2_{s,h}$

Fig. 12. Example 2: snapshots showing the solutions and oscillations of the pressurized elastic membrane at successive times $t \in \{0.05, 0.35, 0.6, 1.3, 1.65, 2.75\}$, respectively. A mesh size $h = 1/160$ is used.

on Strategy I where the convergence of the algorithm at each time step depends on the relative error of the membrane position. In Fig. 13 we report the number of sub-iterations per time step required until the convergence is achieved, for particular values of the fluid viscosity $\mu = 0.025$ and $\mu = 0.075$. The number of sub-iterations is reduced when using smaller values of the viscosity. For $\mu = 0.075$, the converged solution of the algorithm is usually obtained after two sub-iterations, while the Newton algorithm generally converges in two or three sub-iterations.

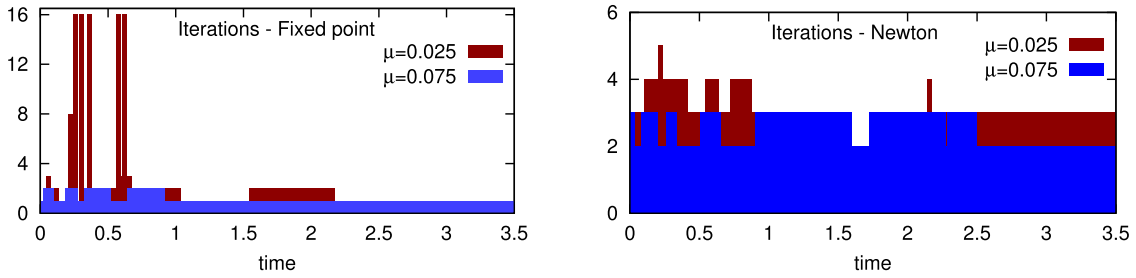


Fig. 13. Example 2: iterations history using the partitioned Strategy I during the first relaxation period of the pressurized membrane.

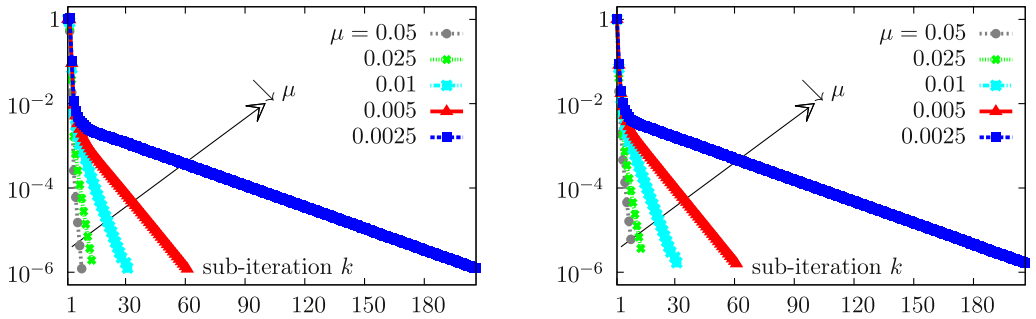


Fig. 14. Example 2: convergence properties of fixed-point iterations using Strategy I (strongly-coupled partitioned approach) for several values of the fluid viscosity. (left) Normalized relative error in φ. (right) Normalized relative error in $\boldsymbol{u}$.

Table 6
Example 2: evaluation of the computing times required by Strategy I and Strategy II for various values of the viscosity.

	CPU time	
μ	Strategy I	Strategy II
0.05	17.150	24.451
0.025	26.415	30.300
0.01	59.029	31.823
0.005	123.929	52.505
0.0025	418.965	68.469

Table 7
Example 2: evaluation of the computing times required by Strategy I and Strategy II for various values of the stiffness.

	CPU time	
T_0	Strategy I	Strategy II
1	10.861	18.511
2	17.150	24.451
3	27.730	29.097
4	53.027	30.555
5	505.603	32.410

We now focus on the influence of the fluid viscosity on the convergence of the strongly-coupled partitioned strategy. We set $\Delta t = 0.01$ and $T_0 = 2$, and we observe the convergence of the sub-iterations with respect to an increasing value of μ. Fig. 14 depicts the evolution of the relative error in the membrane position and fluid velocity. The errors are adimensionalized. We clearly observe that Strategy I slowly converges when decreasing the viscosity, yielding larger computational cost. The computing time required by Strategy I and Strategy II is evaluated in Table 6, showing that the fully monolithic approach is more beneficial for small viscosities.

The influence of the structural stiffness is also worth to be investigated. We proceed with a similar study for fixed values of $\Delta t = 0.01$ and $\mu = 0.05$. Fig. 15 shows that Strategy I is more beneficial than Strategy II for small values of the structural stiffness. However, a very slow linear convergence behavior is observed for large values of T_0 when using the partitioned approach. When increasing the stiffness, Table 7 shows that the fully monolithic strategy allows significant computational savings compared to Strategy I.

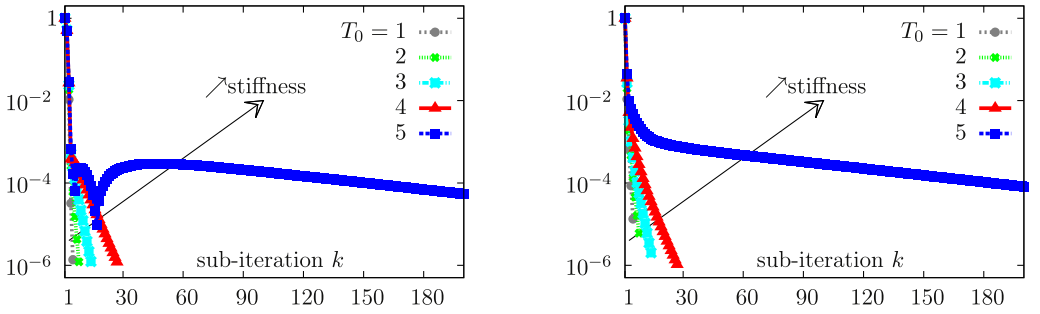


Fig. 15. Example 2: convergence properties of fixed-point iterations using Strategy I (strongly-coupled partitioned approach) for several values of the stiffness. (left) Normalized relative error in φ. (right) Normalized relative error in $\boldsymbol{u}$.

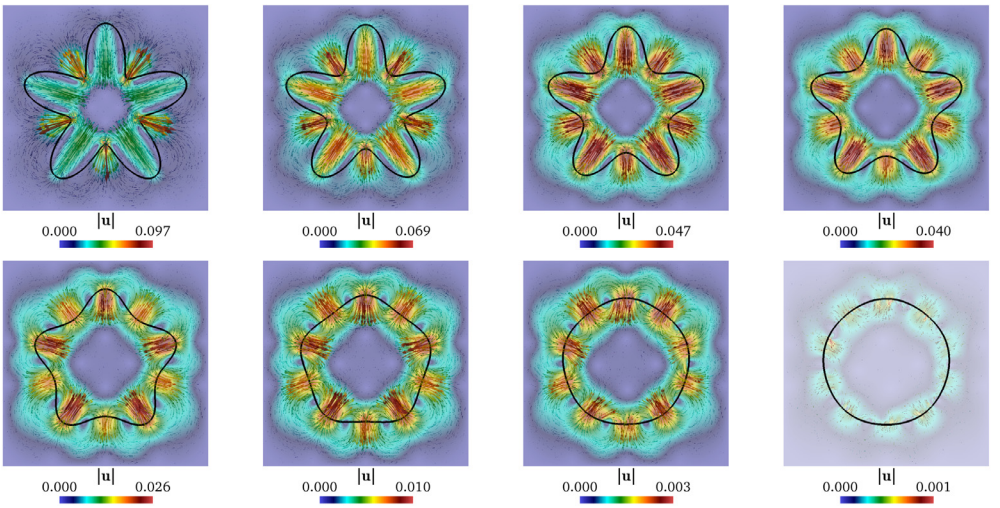


Fig. 16. Example 3: snapshots showing the relaxation of the flower-shape membrane Γ and the velocity profile at successive times $t \in \{0.15, 0.6, 1.2, 1.95, 3.6, 7.2, 12, 14.25\}$, respectively from left to right and from top to bottom.

4.3. Example 3: elastic membrane with a complex initial shape

We now perform a purely Eulerian simulation of the relaxation of a membrane having a more complicated configuration and immersed in a viscous flow. In this example, we do not intend to provide a detailed investigation of the method but instead we only present the shape, velocity and pressure solutions. We consider the test case described in [61,65]. We consider a linear stress-strain relationship. The membrane has initially a flower-like shape which is expressed in the polar coordinates by the following expression:

$$r(\theta) = R_0(1 + \epsilon \sin(\kappa\theta)), \quad \text{with } \theta \in [0, 2\pi],$$

where R_0, ϵ and κ are constants, and κ is an integer setting the number of branches in the shape. We use the same parameters in [61]: $R_0 = 0.5$, $\epsilon = 0.4$ and $\kappa = 5$, while the membrane tension is equal to $T_0 = 0.05$. The membrane is stretched at the initial time. The resting configuration that would be obtained if a small leak across the membrane was permitted is a circle having the radius $r = 0.3$. The velocity and pressure fields are initially set to zero, and the membrane dynamics is driven by the elastic force. The membrane should relax toward an equilibrium shape having the same enclosed area as the initial shape $\pi R_0^2\left(1 + 0.5\epsilon^2\right)$. The computational domain is $[-0.85, 0.85] \times [-0.85, 0.85]$. Following [65], we set different values for the dynamic viscosities on both sides of Γ, which are taken to be $\mu_o = 1$ and $\mu_i = 0.1$, respectively. Computations are performed using a regular mesh having 49′092 elements (i.e. $h = 1/160$). Since the problem is very stiff compared to Example 2 due to the larger curvature compared to Example 2, we first set the time step size to $\Delta t = 9 \times 10^{-3}$. We subsequently use larger time steps, for $t > 5$.

In Fig. 16 we provide snapshots depicting the dynamics of the membrane for $t \in (0, 40)$. The snapshots in Fig. 17 show the pressure distribution at particular times; they clearly demonstrate that the ability of the method to capture the high pressure discontinuities across the membrane. A robust redistancing solver is required to avoid that the discontinuities in the vicinity of the level set skeleton deteriorate the computational accuracy on the membrane, see [46]. In Fig. 18 we present successive snapshots showing the level set solutions at particular times. The temporal evolution of the enclosed

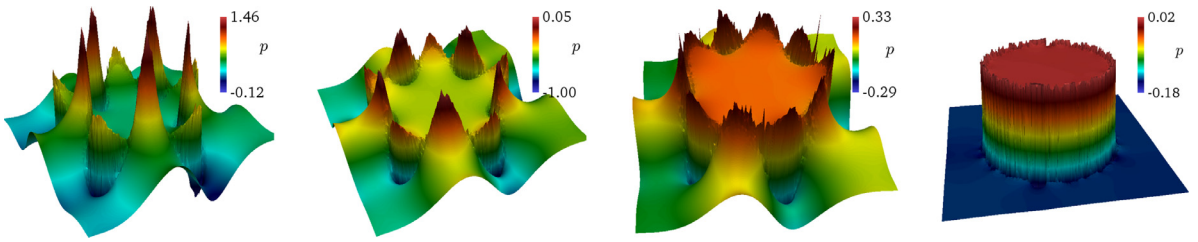


Fig. 17. Example 3: pressure distribution and jump across Γ at successive times $t \in \{1.5, 2.85, 5.4, 25.5\}$, respectively.

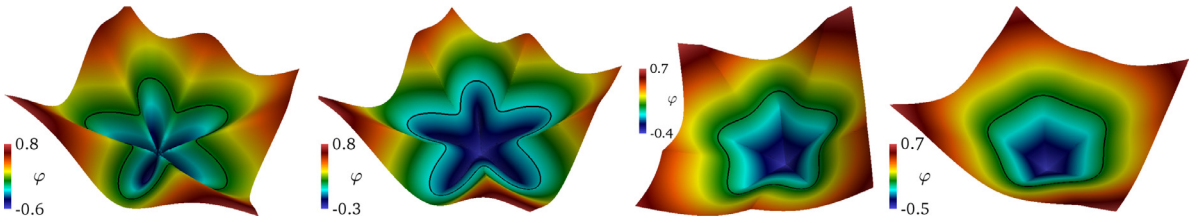


Fig. 18. Example 3: the level set solution at successive times $t \in \{0.15, 0.45, 3.75, 6.6\}$, respectively.

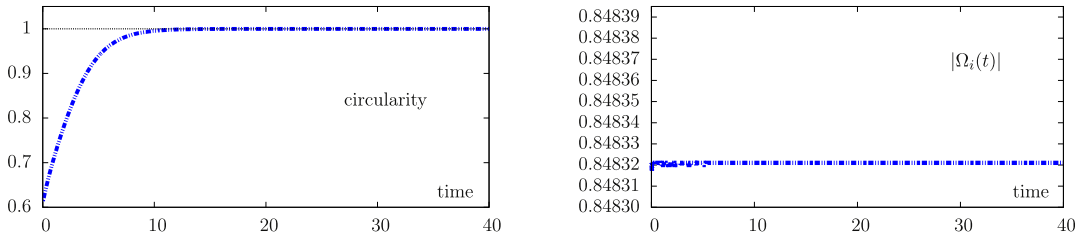


Fig. 19. Example 3: temporal evolution of the circularity (left) and area (right) enclosed by the membrane.

area is plotted in Fig. 19 (right) and illustrates well the preservation of the area during the relaxation process. The final stretched configuration at the steady state is fully circular with a circularity $\not\in(t)$ approaching 1, see Fig. 19 (left).

4.4. Example 4: deformation of three-dimensional elastic membrane

In this final example, we address a three-dimensional configuration where we describe the full membrane elasticity using the neo Hookean model. We consider a spherical unstressed sphere of radius $r = 0.6$ enclosed in the computational domain $[-1.5, 1.5]^3$. We set the material data $\mathrm{E}_s = 20$ and the fluid parameters $\rho_i = \rho_o = 1$ and $\mu_i = \mu_o = 0.3$. The fluid is initially stationary and we prescribe homogeneous Dirichlet boundary conditions for the velocity. A loading force in opposite directions is applied to the membrane in the z-direction for times $t \in [0, 1]$ and is given by the following expression:

$$\boldsymbol{f}(x, y, z) = c\mathscr{H}_\varepsilon\left(x^2 + y^2 - 0.3\right)\mathcal{S}_\varepsilon(z)\delta_\varepsilon(\varphi)\boldsymbol{z}, \quad \text{with } c > 0.$$

The membrane is relaxed afterwards, i.e. for time $t > 1$. At any time $t \in (0, T)$, we introduce the degree of sphericity $\not(t) = \frac{\sqrt[3]{36\pi|\Omega_i(t)|^2}}{|\Gamma(t)|} \in [0, 1]$ as the surface area of a sphere having the same enclosed volume as the deformed sphere at time t. We set $c = 6.5$. Remark that $\not = 1$ corresponds to a perfect sphere and described the fully relaxed membrane. The external force enables to compress the membrane gradually and the maximum deformation is achieved at $t = 1$ where $\not \approx 0.936$, see Fig. 21. Snapshots in Fig. 20 show that the membrane stress during the compression is balanced by the increase in the pressure jump across the membrane. At $t = 1$, the compression force is released and the elastic membrane is relaxed until recovering the perfect spherical shape. The method features good volume preservation, that is, the maximum relative error against the exact volume is about 3.49×10^{-7} at $t = 15$. The time evolutions of the enclosed volume and sphericity degree are reported in Fig. 21. The energy conservation is of great importance. At any time t, $(0, t) \times \Omega$ is denoted by Ω_T. Let us introduce the kinetic energy $\mathscr{E}_\rho$, the viscous energy $\mathscr{E}_\mu$, the elastic energy $\mathscr{E}_\mathrm{W}$ and the work of external forces $\mathscr{E}_{ext}$ as follows:

$$\mathscr{E}_\rho = \frac{1}{2}\int_\Omega \rho\|\boldsymbol{u}\|^2, \quad \mathscr{E}_\mu = \int_{\Omega_T} \mu\|\boldsymbol{\nabla u}\|^2, \quad \mathscr{E}_\mathrm{W} = \int_\Gamma \mathcal{W}(\mathbf{F}_s) \quad \text{and} \quad \mathscr{E}_{ext} = -\int_{\Omega_T} \boldsymbol{f}.\boldsymbol{u}.$$

The total energy $\mathscr{E}_{tot}$ is identified as the summation of the previous energies. Fig. 22 shows the temporal evolution of the different energies and allows to bring to light the good preservation of the total energy throughout the simulation period.

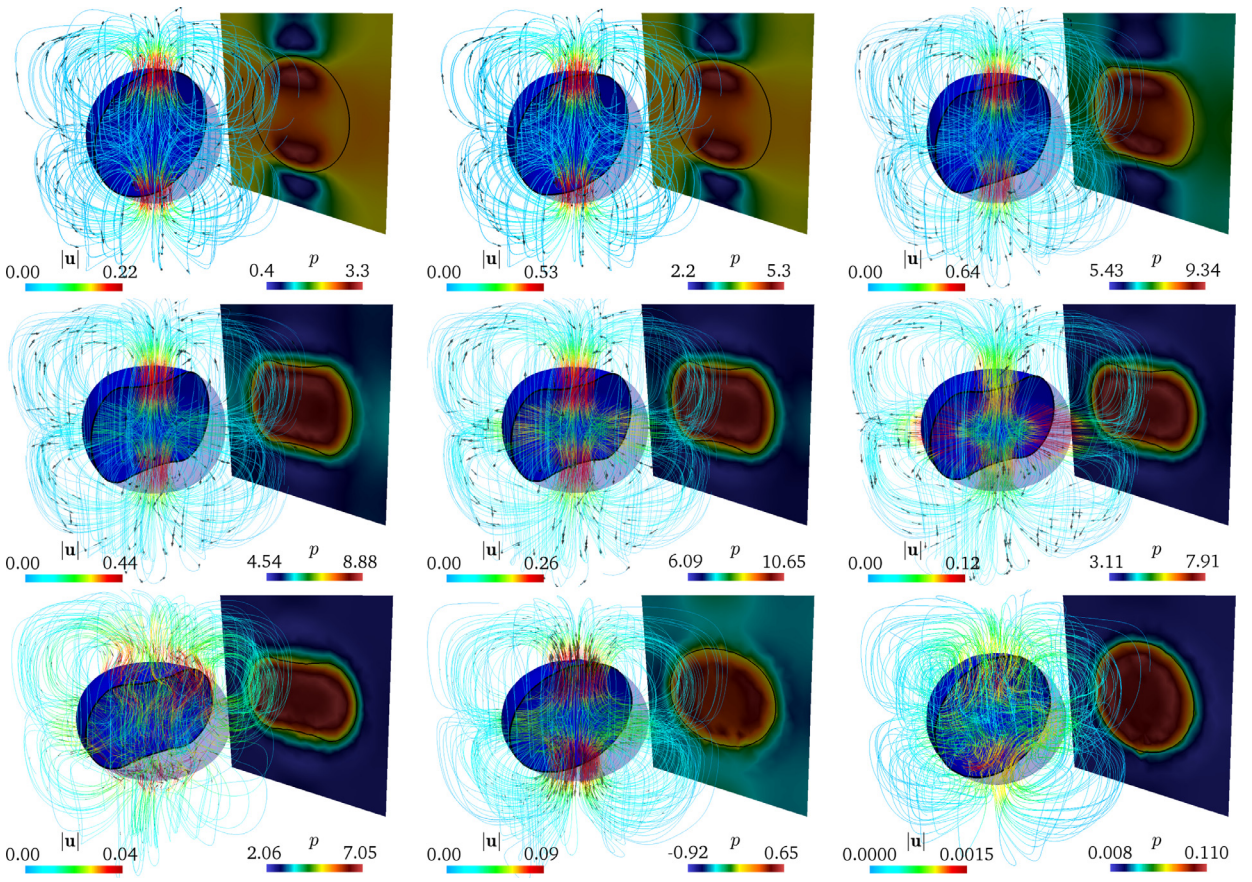


Fig. 20. Example 4: snapshots showing the deformations of the membrane Γ and the pressure profile (plane $x = 0$) at successive times $t \in \{0.03, 0.15, 0.36, 0.54, 0.66, 0.9, 2.4, 8.76\}$, respectively from left to right and from top to bottom.

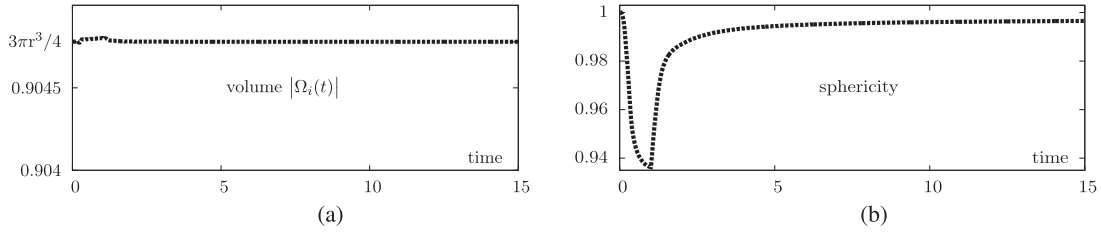


Fig. 21. Example 4: temporal evolution of the enclosed volume (a) and degree of sphericity (b).

In fact, the elastic energy increases during the time interval [0,1] when compression is applied and it allows to balance the external force $\boldsymbol{f}$. When the external force is released, the membrane elastic energy gradually decreases and vanishes when the unstressed spherical shape is reached.

Comparison between the monolithic and partitioned strongly coupled strategies

In this paragraph, we provide further insights into the performances of Strategy I compared to Strategy II in terms of computational cost. Recalling that for both strategies, the coupled problem $\mathscr{P}$ (2.4a)–(2.4f) is solved in an implicit manner, while Strategy I represents a strongly coupled time integration approach that requires an iterative process between the membrane position and the remaining problem. Although the approaches are fully implicit and are unconditionally stable, we cannot use in practice very large time steps that makes the difference between the initial guess and the expected solution too large causing the Newton's method to diverge. In what follows, the time step size is set $\Delta t = 0.01$ and we perform computations in the time interval $(0, T) = (0, 2)$.

The mechanical properties are given the neo-Hookean elasticity model and we consider the same set-up described above. We consider the physical parameters: $c = 10$, $\rho_i = 0.5$, $\rho_o = 1$ and $\mu_i = \mu_o = 0.1$. Let us focus on the computational efforts required by Strategy I and Strategy II for several values of the material stiffness. We measure the wall clock times using the

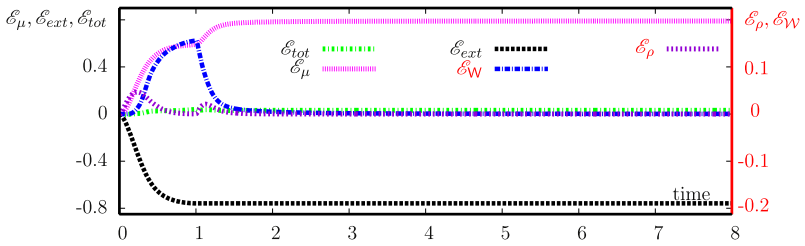


Fig. 22. Example 4: energy conservation and temporal evolution of the different energies.

Table 8
Example 4: evaluation of the computing times required by Strategy I and Strategy II with respect to the stiffness coefficient.

	CPU time					
	$E_s = 0.1$		$E_s = 1$		$E_s = 4$	
#DOFs	Strategy I	Strategy II	Strategy I	Strategy II	Strategy I	Strategy II
103′050	2 h 57 min 25 s	3 h 53 min 19 s	4 h 43 min 39 s	5 h 36 min 47 s	13 h 14 min 58 s	10 h 55 min
255′598	12 h 30 min 29 s	19 h 43 min 33 s	23 h 0 min 31 s	28 h 54 min 19 s	50 h 44 min 26 s	47 h 35 min 42 s
476′556	38 h 55 min 41 s	53 h 57 min 6 s	108 h 14 min 16 s	115 h 18 min 15 s	184 h 0 min 11 s	171 h 6 min 40 s

same number of processors, set equal to 8 in these experiments. We notice that we do not intend to study in this work the parallel performances of the code when increasing the CPU counts; however, we only focus on the influence of the material properties on the computational cost. The computing times CPU are reported in Table 8 using several tetrahedral meshes.

As expected, the performances of both strategies depend on the physical parameters. Numerical results show that Strategy I is computationally cheaper than Strategy II when considering small values of the parameter E_s (typically E_s smaller than 1). That is mainly due to the significantly larger size of the Jacobian matrices when considering Strategy II. However, the complexity of the underlying problem increases when considering larger values of E_s. In such a case, Strategy I exhibits significantly higher number of fixed-point iterations until reaching the convergence and results in substantially higher computational burden. Due to this slow convergence, the monolithic approach becomes preferable from a computational point of view. Nevertheless, we pinpoint that it is necessary to investigate some inexact Newton strategies which can help to reduce importantly the computational cost of both strategies. That represents the content of a forthcoming work we are currently developing.

5. Concluding remarks

This paper presents a computational framework for the modeling of the dynamics of elastic membranes in an incompressible flow. The Eulerian description has been achieved by transporting a modified surface left Cauchy–Green strain tensor, and the method is based on the use of level set and Newton methods in a finite element framework. Since the membrane is assumed massless, resulting in an infinite added mass effect, we have only considered fully implicit time integration schemes. We have devised a strongly-coupled partitioned strategy and a fully monolithic solution algorithm to solve the two-way fluid/membrane coupled problem. Both strategies have used the Kou's method to improve the convergence of the standard Newton method, yielding a third-order convergence behavior without evaluating higher order derivatives. The robustness of the method has been demonstrated through a number of numerical examples. Accuracy tests suggest that the numerical discretization yields a suboptimal convergence behavior, due to the regularization procedure and the inability of the approximation spaces to accurately capture the discontinuities across the membrane. Numerical investigations have brought to light that the partitioned approach is more beneficial in terms of computational cost when using small values of the material stiffness or large values of the fluid viscosity. However, when considering relatively large stiffness or small fluid viscosities, the partitioned approach becomes very inefficient as the fixed point algorithm requires a large number of iterations to satisfy the strong coupling between the subproblems or even fails to converge. In such cases, it is preferable to adopt Strategy II that features substantial computational savings.

Nevertheless, the present method needs several improvements, and we highlight some of the straightforward extensions and applications. As aforementioned, we are currently working on the development of an efficient variant to address the suboptimal convergence behavior of the present method. It is based on both the adaptive Galerkin finite elements and the enrichment of the finite element basis functions to accurately capture the discontinuities. At the numerical level, further developments concerns the construction of robust preconditioners for iterative solvers and inexact Newton strategies. That consists in considering suitable approximations of the Jacobian system, which allow to avoid the recurrent assembly and factorization of the Jacobian matrix. In addition, an ongoing work concerns the development of operator splitting strategies for the time discretization. At the modeling level, we are currently adapting the presented method to the simulation of the hemodynamics in aorta, while modeling the mechanics of the aortic valve and the fibrous structure of leaflets. Another

crucial difficulty concerns the modeling of the contact between leaflets when the valve closes. We foresee to consider the material anisotropy in the leaflets' tangential planes and the mechanical activation in a subsequent step. Finally, this work is part of larger developments to model RBCs under highly nonlinear bending forces and equipped with the hyperelastic thin cytoskeleton in small capillaries.

Acknowledgments

We gratefully acknowledge the financial support by the Swiss National Science Foundation through the grant no. 320030–149567.

References

[1] J. Kou, Y. Li, X. Wang, A modification of Newton method with third-order convergence, Appl. Math. Comput. 181 (2) (2006) 1106–1111.
[2] S.A. Safran, Statistical Thermodynamics of Surfaces, Interfaces and Membranes, Frontier in Physics, Vol 90, Addison-Wesley Publishing Company, Reading, MA, 1994.
[3] A. Hasan, K. Ragaert, W. Swieszkowski, S. Selimovic, A. Paul, G. Camci-Unal, M. Mofrad, A. Khademhosseini, Biomechanical properties of native and tissue engineered heart valve constructs, J. Biomech. 47 (9) (2014) 1949–1963. Functional Tissue Engineering
[4] S. Keller, R. Skalak, Motion of a tank-treading ellipsoidal particle in a shear flow, J. Fluid Mech. 120 (1982) 27–47.
[5] C. Pozrikidis, Numerical simulation of the flow-induced deformation of red blood cells, Ann. Biomed. Eng. 31 (2003) 1194–1205.
[6] A. Mol, N. Driessen, M. Rutten, S. Hoerstrup, C. Bouten, F. Baaijens, Tissue engineering of human heart valve leaflets: a novel bioreactor for a strain-based conditioning approach, Ann. Biomed. Eng. 33 (12) (2005) 1778–1788.
[7] J.W. Barrett, H. Garcke, R. Nürnberg, Numerical computations of the dynamics of fluidic membranes and vesicles, Phys. Rev. E 92 (2015) 052704.
[8] B. Griffith, Immersed boundary model of aortic heart valve dynamics with physiological driving and loading conditions, Int. J. Numer. Methods Biomed. Eng. 28 (3) (2012) 317–345.
[9] Y. Sahasakul, W. Edwards, J. Naessens, A. Tajik, Age-related changes in aortic and mitral valve thickness: Implications for two-dimensional echocardiography based on an autopsy study of 200 normal human hearts, Am. J. Cardiol. 62 (7) (1988) 424–430.
[10] A. Laadhari, G. Székely, Eulerian finite element method for the numerical modeling of fluid dynamics of natural and pathological aortic valves, J. Comput. Appl. Math. 319 (2017) 236–261.
[11] A. Laadhari, Numerical modelling of the dynamics of red blood cells using the level set method, Doctoral School Mathematics, Information Sciences and Technologies, and Computer Science, Grenoble, 2011 Ph.D. thesis. http://www.theses.fr/2011GRENM011
[12] J. Barrett, H. Garcke, R. Nürnberg, A stable numerical method for the dynamics of fluidic membranes, Numer. Math. 134 (4) (2016) 783–822.
[13] S.C. Vigmostad, H.S. Udaykumar, J. Lu, K.B. Chandran, Fluid-structure interaction methods in biological flows with special emphasis on heart valve dynamics, Int. J. Numer. Methods Biomed. Eng. 26 (3–4) (2010) 435–470.
[14] C.S. Peskin, Flow patterns around heart valves: a numerical method, J. Comput. Phys. 10 (2) (1972) 252–271.
[15] B. Griffith, Simulating the blood-muscle-valve mechanics of the heart by an adaptive and parallel version of the immersed boundary method, New York University, 2005 PhD Thesis.
[16] B. Griffith, C. Peskin, On the order of accuracy of the immersed boundary method: higher order convergence rates for sufficiently smooth problems, J. Comput. Phys. 208 (1) (2005) 75–105.
[17] B. Griffith, On the volume conservation of the immersed boundary method, Commun. Comput. Phys. 12 (2012) 401–432.
[18] Y. Kim, M.-C. Lai, Simulating the dynamics of inextensible vesicles by the penalty immersed boundary method, J. Comput. Phys. 229 (12) (2010) 4840–4853.
[19] W.-F. Hu, K.Y. Kim, M.-C. Lai, An immersed boundary method for simulating the dynamics of three-dimensional axisymmetric vesicles in Navier–Stokes flows, J. Comput. Phys. 257, Part A (2014) 670–686.
[20] M.-C. Lai, C.S. Peskin, An immersed boundary method with formal second-order accuracy and reduced numerical viscosity, J. Comput. Phys. 160 (2) (2000) 705–719.
[21] D.-V. Le, Z. Tan, Large deformation of liquid capsules enclosed by thin shells immersed in the fluid, J. Comput. Phys. 229 (11) (2010) 4097–4116.
[22] Y. Seol, W.-F. Hu, Y. Kim, M.-C. Lai, An immersed boundary method for simulating vesicle dynamics in three dimensions, J. Comput. Phys. 322 (2016) 125–141.
[23] G.-H. Cottet, E. Maitre, A level-set formulation of immersed boundary methods for fluid-structure interaction problems, Comptes Rendus Math. 338 (7) (2004) 581–586.
[24] M.-C. Lai, Z. Li, A remark on jump conditions for the three-dimensional Navier-Stokes equations involving an immersed moving membrane, Appl. Math. Lett. 14 (2) (2001) 149–154.
[25] F.P.T. Baaijens, A fictitious domain/mortar element method for fluid-structure interaction, Int. J. Numer. Methods Fluids 35 (7) (2001) 743–761.
[26] T. Richter, T. Wick, Finite elements for fluid-structure interaction in ale and fully Eulerian coordinates, Comput. Methods Appl. Mech. Eng. 199 (2010) 2633–2642.
[27] A. Laadhari, R. Ruiz-Baier, A. Quarteroni, Fully Eulerian finite element approximation of a fluid-structure interaction problem in cardiac cells, Int. J. Numer. Methods Eng. 96 (11) (2013) 712–738.
[28] C. Pozrikidis, S. Ramanujan, Deformation of liquid capsules enclosed by elastic membranes in simple shear flow: large deformations and the effect of fluid viscosities, J. Fluid Mech. 361 (1998) 117–143.
[29] A. Rahimian, S.K. Veerapaneni, G. Biros, Dynamic simulation of locally inextensible vesicles suspended in an arbitrary two-dimensional domain, a boundary integral method, J. Comput. Phys. 229 (18) (2010) 6466–6484.
[30] F. Vernerey, M. Farsad, An Eulerian/XFEM formulation for the large deformation of cortical cell membrane, Comput. Methods Biomech. Biomed. Eng. 14 (5) (2011) 433–445.
[31] D. Boffi, N. Cavallini, L. Gastaldi, Finite element approach to immersed boundary method with different fluid and solid densities, Math. Models Methods Appl. Sci. 21 (12) (2011) 2523–2550.
[32] G.-H. Cottet, E. Maitre, A semi-implicit level set method for multiphase flows and fluid-structure interaction problems, J. Comput. Phys. 314 (2016) 80–92.
[33] F. Gibou, C. Min, Efficient symmetric positive definite second-order accurate monolithic solver for fluid/solid interactions, J. Comput. Phys. 231 (8) (2012) 3246–3263.
[34] E. Ramm, W.A. Wall, Shell structures-a sensitive interrelation between physics and numerics, Int. J. Numer. Methods Eng. 60 (1) (2004) 381–427.
[35] J. Banks, W. Henshaw, D. Schwendeman, An analysis of a new stable partitioned algorithm for {FSI} problems. part ii: Incompressible flow and structural shells, J. Comput. Phys. 268 (2014) 399–416.
[36] J. Kou, Y. Li, X. Wang, Third-order modification of Newton's method, J. Comput. Appl. Math. 205 (1) (2007) 1–5.
[37] M. Gurtin, A. Murdoch, A continuum theory of elastic material surfaces, Arch. Rational Mech. Anal. 57 (4) (1975) 291–323.
[38] D. Barthès-Biesel, J. Rallison, The time-dependent deformation of a capsule freely suspended in a linear shear flow, J. Fluid Mech. 113 (1981) 251–267.
[39] X. Li, D. Barthes-Biesel, A. Helmy, Large deformations and burst of a capsule freely suspended in an elongational flow, J. Fluid Mech. 187 (1988) 179–196.

[40] A. Laadhari, A. Quarteroni, Numerical modeling of heart valves using resistive Eulerian surfaces, Int. J. Numer. Methods Biomed. Eng. 32 (5) (2016) e02743–n/a.
[41] P. Saramito, Efficient C++ finite element computing with Rheolef, CNRS-CCSD, 2013. www-ljk.imag.fr/membres/Pierre.Saramito/rheolef, accessed: 2018-03-13
[42] P. Gomez, J. Hernandez, J. Lopez, On the reinitialization procedure in a narrow-band locally refined level set method for interfacial flows, Int. J. Numer. Methods Eng. 63 (10) (2005) 1478–1512.
[43] D. Hartmann, M. Meinke, W. Schröder, The constrained reinitialization equation for level set methods, J. Comput. Phys. 229 (5) (2010) 1514–1535.
[44] A. Laadhari, P. Saramito, C. Misbah, Improving the mass conservation of the level set method in a finite element context, Comptes Rendus Math. 348 (9) (2010) 535–540.
[45] A. Laadhari, P. Saramito, C. Misbah, Computing the dynamics of biomembranes by combining conservative level set and adaptive finite element methods, J. Comput. Phys. 263 (0) (2014) 328–352.
[46] A. Laadhari, P. Saramito, C. Misbah, An adaptive finite element method for the modeling of the equilibrium of red blood cells, Int. J. Numer. Methods Fluids 80 (7) (2016) 397–428.
[47] P. Saramito, Complex fluids, modeling and algorithms, Mathmatiques et Applications, 79, Springer International Publishing Switzerland, 2016.
[48] S. Weerakoon, T. Fernando, A variant of Newton's method with accelerated third-order convergence, Appl. Math. Lett. 13 (8) (2000) 87–93.
[49] A. Quarteroni, F.S. R. Sacco, Numerical mathematics, Comput. Methods Appl. Mech. Eng. 37 (2000).
[50] E.H. van Brummelen, Partitioned iterative solution methods for fluid-structure interaction, Int. J. Numer. Methods Fluids 65 (1–3) (2011) 3–27.
[51] A. Laadhari, P. Saramito, C. Misbah, G. Székely, Fully implicit methodology for the dynamics of biomembranes and capillary interfaces by combining the level set and newton methods, J. Comput. Phys. 343 (2017) 271–299.
[52] L. Lee, R.J. LeVeque, An immersed interface method for incompressible Navier–Stokes equations, SIAM J. Sci. Comput. 25 (2003) 832–856.
[53] S. Hysing, S. Turek, D. Kuzmin, N. Parolini, E. Burman, S. Ganesan, L. Tobiska, Quantitative benchmark computations of two-dimensional bubble dynamics, Int. J. Numer. Methods Fluids 60 (11) (2009) 1259–1288.
[54] L. Sťrubelj, I. Tiselj, B. Mavko, Simulations of free surface flows with implementation of surface tension and interface sharpening in the two-fluid model, Int. J. Heat Fluid Flow 30 (4) (2009) 741–750.
[55] J. Klostermann, K. Schaake, R. Schwarze, Numerical simulation of a single rising bubble by VOF with surface compression, Int. J. Numer. Methods Fluids. 71 (8) (2013) 960–982.
[56] A. Laadhari, G. Székely, Fully implicit finite element method for the modeling of free surface flows with surface tension effect, Int. J. Numer. Methods Eng. 111 (11) (2017) 1047–1074. Nme.5493.
[57] V. Doyeux, Y. Guyot, V. Chabannes, C. Prud'homme, M. Ismail, Simulation of two-fluid flows using a finite element/level set method. Application to bubbles and vesicle dynamics, J. Comput. Appl. Math. 246 (2013) 251–259.
[58] D. Le, An immersed interface method for solving viscous incompressible flows involving rigid and flexible boundaries, Singapore-MIT Alliance, 2005 Ph.D. Thesis.
[59] Z. Tan, D. Le, Z. Li, K. Lim, B. Khoo, An immersed interface method for solving incompressible viscous flows with piecewise constant viscosity across a moving elastic membrane, J. Comput. Phys. 227 (23) (2008) 9955–9983.
[60] W.-P. Breugem, A second-order accurate immersed boundary method for fully resolved simulations of particle-laden flows, J. Comput. Phys. 231 (13) (2012) 4469–4498.
[61] Z. Li, M.-C. Lai, The immersed interface method for the Navier–Stokes equations with singular forces, J. Comput. Phys. 171 (2) (2001) 822–842.
[62] J. Li, J.M. Melenk, B. Wohlmuth, J. Zou, Optimal a priori estimates for higher order finite elements for elliptic interface problems, Appl. Numer. Math. 60 (1–2) (2010) 19–37.
[63] D.R. Noble, E.P. Newren, J.B. Lechman, A conformal decomposition finite element method for modeling stationary fluid interface problems, Int. J. Numer. Methods Fluids 63 (6) (2010) 725–742.
[64] T.-P. Fries, T. Belytschko, The extended/generalized finite element method: an overview of the method and its applications, Int. J. Numer. Methods Eng. 84 (3) (2010) 253–304.
[65] Z. Tan, D. Le, Z. Li, K. Lim, B. Khoo, An immersed interface method for solving incompressible viscous flows with piecewise constant viscosity across a moving elastic membrane, J. Comput. Phys. 227 (23) (2008) 9955–9983.