

# A particle-based moving interface method (PMIM) for modeling the large deformation of boundaries in soft matter systems

Louis Foucard and Franck J. Vernerey<sup>\*,†</sup>

*Department of Civil, Environmental and Architectural Engineering, University of Colorado, Boulder, CO, USA*

## SUMMARY

The mechanics of the interaction between a fluid and a soft interface undergoing large deformations appear in many places, such as in biological systems or industrial processes. We present an Eulerian approach that describes the mechanics of an interface and its interactions with a surrounding fluid via the so-called Navier boundary condition. The interface is modeled as a curvilinear surface with arbitrary mechanical properties across which discontinuities in pressure and tangential fluid velocity can be accounted for using a modified version of the extended finite element method. The coupling between the interface and the fluid is enforced through the use of Lagrange multipliers. The tracking and evolution of the interface are then handled in a Lagrangian step with the grid-based particle method. We show that this method is ideal to describe large membrane deformations and Navier boundary conditions on the interface with velocity/pressure discontinuities. The validity of the model is assessed by evaluating the numerical convergence for a axisymmetrical flow past a spherical capsule with various surface properties. We show the effect of slip length on the shear flow past a two-dimensional capsule and simulate the compression of an elastic membrane lying on a viscous fluid substrate. Copyright © 2015 John Wiley & Sons, Ltd.

Received 16 September 2013; Revised 21 September 2015; Accepted 5 December 2015

KEY WORDS: membrane/fluid interactions; Eulerian; extended finite element method

## 1. INTRODUCTION

The problem of the interactions between a fluid and an immersed interface undergoing large shape changes is relevant to many areas of engineering and biological research but still remains a significant computational challenge. Examples of this problem are ubiquitous in systems involving soft matter, including the mechanics of the cell membrane and vesicles and the extreme deformations they are subjected to during migration [1–3], blebbing [4, 5], division [6], and budding [7, 8]. From a computational point of view, one of the most acknowledged method to couple fluid motion and structure deformation is the immersed boundary method [9–11]. The method particularly relies on three features: the fluid flow equations are handled with an Eulerian approach, the membrane deformation are described within a Lagrangian frame, and the fluid–structure interactions are handled via a forcing term that is localized on the membrane domain. It is able to simulate the deformation of thin boundaries as well as three-dimensional structures immersed in an incompressible flow [12, 13]. An approach similar to the immersed boundary method is the distributed Lagrange multiplier/fictitious domain method [14, 15], which has also been adapted to simulate deformable thin surfaces [16, 17]. Here, the coupling between the fluid and the structure is enforced in the entire domain occupied by the structure. The structure is therefore tied to the underlying fictitious flow and has to deform in an incompressible way, which limits the type of materials that can be simulated.

<sup>\*</sup>Correspondence to: Franck J Vernerey, Department of Civil, Environmental and Architectural Engineering, Material Science and Engineering, University of Colorado, 1111 Engineering Drive, 428 UCB, ECOT 422 Boulder, CO 80309-0428, USA.

<sup>†</sup>E-mail: Franck.Vernerey@colorado.edu

Another approach developed to study fluid/interface interaction within the creeping flow regime is the boundary integral method, where only the surface of the interface needs to be discretized. The method is very successful at simulating drops in viscous flows [18, 19] and was extended to elastic interfaces by projecting the velocity gradient of the surrounding fluid to find the deformation rate of the interface [20, 21].

Most of these methods, however, have in common a Lagrangian mesh for the structure interacting with the fluid and cannot easily handle the large deformations or viscoelastic behaviors observed in many biological systems. Indeed, when the membrane is described in a Lagrangian framework, extreme deformations often lead to severe distortions of the finite element mesh, an issue that can only be approached by complicated and computationally expensive mesh regularization techniques [22].

A solution to this problem was presented by Cottet *et al.* in [23] via the introduction of a fully Eulerian description of the system, in which kinematic quantities representing the interface motion (such as position or dilation) are implicitly described by a level-set function, which is defined and updated on a fixed underlying Eulerian grid. The approach has the advantage of removing the need for discretizing the two-dimensional surface and effectively eliminates the issue of mesh distortion. Using this formulation, the authors successfully predicted the equilibrium shape of a red blood cell and showed that large membrane deformation could be naturally handled. Similar level-set approaches are typically used to simulate fluid interfaces and bio-film growth [24, 25] and generalized to elasto-plastic materials [26, 27], as well as hyperelastic materials [28–31]. Two major limitations of existing level-set formulations, however, are as follows. First, a level-set function is typically unable to carry and transport Lagrangian information associated with the interface, such as the interface strain or a concentration of interface particles. The technique is therefore restricted to fluid interfaces or limited cases of elastic behaviors [23]. Second, the level-set evolution equation implicitly assumes that the fluid flow is continuous across the interface ('stick' condition between fluid and interface) and that only the component of the velocity normal to the interface is responsible for its deformation. This precludes the consideration of 'slip' conditions, which are known to be preponderant in many situations such as flows at the microscale or nanoscale [32], on hydrophobic surfaces, or even on hydrophilic surfaces under certain conditions [33].

In an attempt to address the aforementioned shortcomings, we present here an Eulerian formulation that can capture arbitrarily large deformations of an immersed interface, which may be fluid or elastic, with varying surface properties dictated by the Navier boundary conditions. To remain general, the term interface designates here an arbitrary surface separating two fluids, and can be endowed with a wide range of mechanical properties, from surface tension for simple fluid interfaces to more complex elastic behaviors for membranes [34, 35]. Similarly to some recent work on fluid–structure interactions [36, 37], the present formulation uses the extended finite element method (X-FEM) [38], and the coupling between the fluid and the interface is handled with a Lagrange multiplier [39]. The formulation marches towards equilibrium by solving a series of Eulerian and Lagrangian steps, which solves for the fluid and membrane velocities then updates the membrane position and deformation in a staggered fashion. The proposed approach, however, does not use a Lagrangian mesh to discretize the interface and has the following three contributions:

- The model endows the interface with its own independent velocity and curvilinear coordinate system, which is required for the calculation of the surface velocity gradient that dictates the interface deformation, and for the implementation of the Navier boundary condition. By varying the slip length  $l$ , the surface condition at any point on the interface can change continuously from 'slip' to 'stick'.
- The method uses a modified version of the X-FEM introduced by Song *et al.* [40] and adapts it to naturally handles discontinuities in pressure and tangential velocity arising from the Navier boundary condition and the interface elastic stress. The tangential enrichment in the velocity field only imposes the discontinuity in the tangential direction and the continuity condition in the normal direction is automatically satisfied, which eliminates the need for an additional Lagrange multiplier field.

- The X-FEM is used in combination with a grid-based particle method [41] to provide an explicit description of the geometry of the interface and to transport the Lagrangian quantities on the interface. This method of tracking the interface geometry is not subject to mesh distortion and can therefore handle arbitrarily large deformations without needing time-consuming mesh reinitialization algorithms.

We show that the formulation is well adapted for investigating the interactions between fluid and soft matter interfaces and when the effect of surface properties become relevant, which is ideal for a wide range of biological problems. We replicate results from that in [42] for the deformation of two-dimensional elastic capsule in shear flow and further show the effect of slip length on the capsule deformation and ‘tank-threading’ motion. The outline of the paper is as follows: Section 2 introduces the kinematics and deformation measures used for the fluid and the interface in an Eulerian framework, as well as the governing and constitutive equations. The numerical implementation of the derived equation is then discussed in the context of the X-FEM and the grid-based particle method in Section 3. The validity and accuracy of the formulation are then assessed in Section 4, where we consider the numerical convergence for an axisymmetrical flow past a spherical capsule with various surface properties and membrane rigidities and the shear flow past a two-dimensional capsule and the effect of slip length. Our last example shows the folding of an elastic membrane laying on top of a viscous fluid substrate. We first compare our results with the known analytical solution for the membrane profile in the hydrostatic case and then probe the viscous effects on the folding dynamics. Section 5 finally concludes with a discussion of the results and possible future applications.

## 2. MATHEMATICAL FORMULATION FOR A SOFT IMMERSED INTERFACE

### 2.1. Geometry and deformation measures of a membrane in Eulerian framework

Consider a curvilinear surface  $\Gamma$  in a three-dimensional Euclidean space whose geometry is defined by the position of a point  $\mathbf{x}$  through the parameterization  $x^i(t) = r^i(\xi^\alpha, t)$ , where  $\xi^\alpha$ ,  $\alpha = 1, 2$  are the fixed surface coordinates and  $x^i$ ,  $i = 1, 2, 3$  are the components of the position vector. In the remainder of the paper, greek indices will be considered as running on values 1,2, while latin indices will span values 1,2,3, both being summed over their respective range if repeated. From the parameterization  $r^i(\xi^\alpha, t)$ , one can define the components of two vectors  $\mathbf{a}_1$  and  $\mathbf{a}_2$  that form the basis of the plane tangent to  $\Gamma$  at point  $\mathbf{x}$  as  $a^i_\alpha = \partial r^i / \partial \xi^\alpha$ . A third vector  $\bar{\mathbf{n}}$ , orthogonal to the surface and of unit length, is defined as  $\bar{\mathbf{n}} = \mathbf{a}_1 \times \mathbf{a}_2 / |\mathbf{a}_1 \times \mathbf{a}_2|$  and represents the orientation of  $\Gamma$  at point  $\mathbf{x}$ . The induced first fundamental form of the surface  $\Gamma$ , also called metric, is given by the scalar product,  $a_{\alpha\beta} = \mathbf{a}_\alpha \cdot \mathbf{a}_\beta$ , and can be used to define a dual basis to  $\mathbf{a}_\alpha$  as  $\mathbf{a}^\alpha = a^{\alpha\beta} \mathbf{a}_\beta$ , whereby  $a^{\alpha\beta} = (a_{\alpha\beta})^{-1}$  and  $\mathbf{a}_\alpha \cdot \mathbf{a}^\beta = \delta_{\alpha\beta}$ . The second fundamental form is a surface tensor field whose components in the basis of the tangent plane are  $b_{\alpha\beta} := -\mathbf{a}_\alpha \cdot \bar{\mathbf{n}}_{,\beta}$ . An appropriate measure of surface bending, which we will refer to as the curvature tensor  $\bar{\mathbf{C}}$ , can then be obtained by calculating the change in the second fundamental form between the reference and the current configurations as follows:

$$\bar{\mathbf{C}} = (b_{\alpha\beta} - B_{\alpha\beta}) \mathbf{a}^\alpha \otimes \mathbf{a}^\beta = \bar{C}^{\text{ip}}_{\alpha\beta} \mathbf{a}^\alpha \otimes \mathbf{a}^\beta \tag{1}$$

where  $B_{\alpha\beta}$  is the intrinsic interface curvature expressed in the  $\mathbf{a}$  basis and  $\bar{C}^{\text{ip}}_{\alpha\beta} = b_{\alpha\beta} - B_{\alpha\beta}$  is the in-plane curvature tensor. Note that the former identically vanishes for a surface that is intrinsically flat. The first invariant of the second fundamental form  $\mathbf{b}$  yields the mean curvature as  $H = a^{\alpha\beta} b_{\beta\alpha} / 2 = b^\alpha_\alpha / 2$ .

To characterize the in-plane deformation of the interface, let us now introduce the deformation tensor  $\bar{\mathbf{F}}$ :

$$\bar{\mathbf{F}} = \bar{F}^{\text{ip}}_{\beta\alpha} \mathbf{a}^\beta \otimes \mathbf{a}^\alpha \quad \text{with} \quad \bar{F}^{\text{ip}}_{\beta\alpha} = \frac{\partial \xi^\beta}{\partial \Xi^\alpha}, \tag{2}$$

where  $\xi^\beta$  and  $\Xi^\alpha$  denote the curvilinear coordinates of an interface material point in the reference and deformed configuration, respectively. The deformation gradient can be represented in two ways:

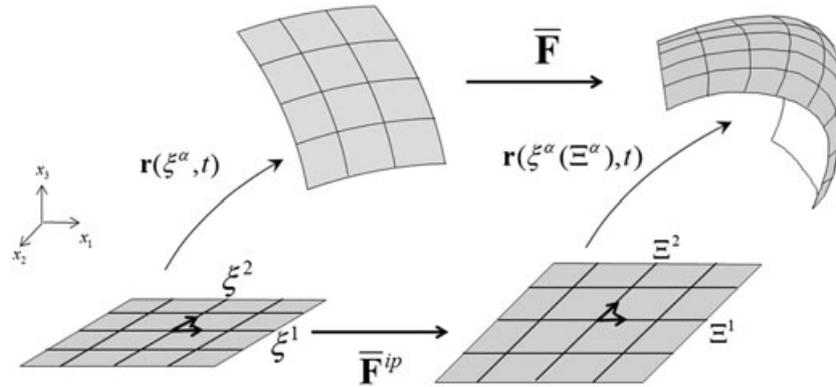


Figure 1. Reference and deformed configuration of the interface. The parametrization  $\mathbf{r}$  maps the two-dimensional space  $\xi^\alpha$  to the 3D space  $x_i$ . The  $(2 \times 2)$  deformation tensor  $\bar{\mathbf{F}}^{ip}$  maps the deformation of a material point  $\xi^\alpha$  in the two-dimensional reference configuration to the two-dimensional deformed configuration  $\Xi^\alpha$ . The  $(3 \times 3)$  deformation tensor  $\bar{\mathbf{F}}$  maps the deformation of a material point  $\mathbf{r}(\xi^1, \xi^2)$  in the three-dimensional reference configuration to the three-dimensional deformed configuration  $\mathbf{r}(\Xi^1(\xi^1, \xi^2), \Xi^2(\xi^1, \xi^2))$ .

(1) by the full deformation gradient  $\bar{\mathbf{F}}$  with dimension  $(3 \times 3)$  and (b) by the in-plane deformation gradient  $\bar{\mathbf{F}}^{ip}$  with dimension  $(2 \times 2)$  (Figure 1). To avoid complication due to rigid body motion, it is more convenient to work with the symmetric part of the deformation gradient, or equivalently, with the Green–Lagrange strain tensor defined as

$$\bar{\mathbf{E}} = \frac{1}{2}(\bar{\mathbf{F}}^T \bar{\mathbf{F}} - \mathbf{I}) = \bar{E}_{\alpha\beta}^{ip} \mathbf{a}^\alpha \otimes \mathbf{a}^\beta \tag{3}$$

where the identity tensor  $\mathbf{I}$  can be written in terms of the reference configuration basis  $\mathbf{I} = a_{\alpha\beta} \mathbf{a}^\alpha \otimes \mathbf{a}^\beta$  and  $\bar{\mathbf{E}}^{ip}$  is the in-plane Green–Lagrange strain.

We now have a complete description of the interface position and deformation. To complete the description, we now turn to the kinematics of the surrounding fluid and its interaction with the interface.

### 2.2. Kinematics of the fluid/interface interactions

Let us now consider the (closed) interface  $\Gamma$  immersed in a fluid domain denoted by  $\Omega$ . Let us further denote as  $\Omega^-$  the subdomain that is enclosed within the interface and  $\Omega^+$  the external subdomain such that  $\Omega = \Omega^- \cup \Omega^+$ . Before we turn to the governing equations for the interface/fluid system, we introduce the operators needed to describe the kinematics of the interaction between the interface and the surrounding fluid. For this, we consider an arbitrary Eulerian vector field  $\mathbf{f}(\mathbf{x}, t)$  (which may be the velocity) and seek a mathematical representation at a point that is very near the interface, whose normal vector at this location is  $\bar{\mathbf{n}}$ . It is convenient to write the vector  $\mathbf{f}$  in terms of its normal and tangential projections on  $\Gamma$  as

$$\mathbf{f} = \mathbf{f}^\perp + \mathbf{f}^\parallel \quad \text{with} \quad \mathbf{f}^\perp = \mathbf{P}^\perp \cdot \mathbf{f} \quad \text{and} \quad \mathbf{f}^\parallel = \mathbf{P}^\parallel \cdot \mathbf{f} \tag{4}$$

where the normal and tangential projection operators are defined by

$$P_{ij}^\perp = \bar{n}^i \bar{n}^j \quad \text{and} \quad P_{ij}^\parallel = a_1^i a_2^j.$$

In addition, to account for the existence of Navier-type boundary conditions at the fluids/interface boundary, one needs to introduce field discontinuities across  $\Gamma$ . Denoting  $\mathbf{v}^-$  and  $\mathbf{v}^+$  as the fluid velocity field  $\mathbf{v}$  in  $\Omega^+$  and  $\Omega^-$ , respectively, it is possible to introduce discontinuities in tangential velocity across  $\Gamma$  as [43]

$$[\mathbf{v}]^{\parallel+} = \bar{\mathbf{v}}^\parallel - \mathbf{v}^{\parallel+}, \quad [\mathbf{v}]^{\parallel-} = \mathbf{v}^{\parallel-} - \bar{\mathbf{v}}^\parallel \quad \text{and} \quad [\mathbf{v}]^\parallel = \mathbf{v}^{\parallel+} - \mathbf{v}^{\parallel-}. \tag{5}$$

Further assuming that no normal separation occurs between the fluids and the interface, we obtain the following normal velocity continuity conditions

$$\mathbf{v}^{\perp+} = \mathbf{v}^{\perp-} = \bar{\mathbf{v}}^{\perp}, \tag{6}$$

where we used the decomposition of the velocity vector in normal and tangential components as defined in (4). With this formalism, the motion at the interface  $\Gamma$  and the nearby fluids are entirely specified by the combination of three independent variables  $\{\bar{\mathbf{v}}, [\mathbf{v}]^{\parallel}, \mathbf{v}^{\parallel-}\}$  or, equivalently,  $\{\bar{\mathbf{v}}, \mathbf{v}^{\parallel+}, \mathbf{v}^{\parallel-}\}$ .

### 2.3. Basic governing equations

Consider a two-dimensional incompressible viscous flow in the domain  $\Omega$  delimited by a boundary  $\partial\Omega$  in which exists a closed vesicle, with surface  $\Gamma$  that is able to move with the surrounding fluid. The problem is characterized by the Reynolds number  $\mathcal{R}_e = HV\rho/\mu$ , where  $H$  is the characteristic length scale,  $V$  the characteristic fluid velocity,  $\mu$  the kinematic viscosity, and  $\rho$  the fluid densities in and out of the vesicle. We choose here to remain in the Stokes flow assumption with  $\mathcal{R}_e \ll 1$ , where inertial effect may be neglected. The velocity of a fluid particle is given in terms of its material time derivative  $\mathbf{v}(\mathbf{x}, t) = D\mathbf{x}/Dt$ , where  $\mathbf{x}$  is the current position of the fluid particle at time  $t$ . Under these conditions, the governing equations for the Stokes flow are written as

$$\nabla \cdot \boldsymbol{\sigma} = \mathbf{0} \quad \forall \mathbf{x} \in \Omega / \Gamma \tag{7}$$

$$\nabla \cdot \mathbf{v} = 0 \quad \forall \mathbf{x} \in \Omega / \Gamma \tag{8}$$

where  $\boldsymbol{\sigma}$  is the Cauchy stress tensor in the fluid and the second equation imposes the condition of incompressibility. These equations are combined with the moving interface problem:

$$[\boldsymbol{\sigma} \cdot \bar{\mathbf{n}}] = \bar{\mathbf{f}} \quad \forall \mathbf{x} \in \Gamma \tag{9}$$

$$(\boldsymbol{\sigma} \cdot \bar{\mathbf{n}})^{\parallel\pm} = \frac{\mu}{l^{\pm}} [\mathbf{v}]^{\parallel\pm} \quad \forall \mathbf{x} \in \Gamma \tag{10}$$

$$\frac{d\mathbf{r}(\xi^1, \xi^2, t)}{dt} = \bar{\mathbf{v}}(\xi^1, \xi^2, t) \tag{11}$$

with the force  $\mathbf{f}$  is the unbalanced interface force due to its deformation. Equation (10) is the Navier boundary conditions at the interface, where  $l^{\pm} > 0$  is known as the slip length, a physical parameter that depends on the surface roughness. The case  $l = 0$  corresponds to the no-slip condition, while  $l = \infty$  is the free slip condition between the fluid and the interface. Finally, the boundary conditions for fluid motion and pressure at the external boundary read

$$\boldsymbol{\sigma} \cdot \mathbf{n} = p_0 \mathbf{n} \quad \forall \mathbf{x} \in \partial\Omega_p \tag{12}$$

$$\mathbf{v}(\mathbf{x}, t) = \mathbf{v}_0 \quad \forall \mathbf{x} \in \partial\Omega_v, \tag{13}$$

where  $p_0$  is an external pressure surrounding the domain  $\Omega$  and a Dirichlet-type condition can be applied on the velocity on the subdomain  $\partial\Omega_v$ .

### 2.4. Constitutive equations

To complement the aforementioned system of equation, a number of constitutive relations must be introduced. They can be broken down into two components that describe in turn the (a) behavior of the fluid and (b) the mechanical behavior of the interface. In this work, we consider a simple incompressible Newtonian fluid with viscosity  $\mu$ , which yields the familiar viscous stress tensor

$$\boldsymbol{\sigma} = \mu \mathbf{D} - p \mathbf{I}, \tag{14}$$

where  $\mathbf{D}$  is the rate of deformation and  $p$  is the hydrostatic pressure. In the case of an elastic thin membrane, a strain energy function  $\bar{\Phi} = \bar{\Phi}(\bar{\mathbf{C}}^{\text{ip}}, \bar{\mathbf{E}}^{\text{ip}})$  can be generally defined in terms of the curvature  $\bar{\mathbf{C}}^{\text{ip}}$  and strain  $\bar{\mathbf{E}}^{\text{ip}}$  introduced in the previous section. The in-plane Cauchy stress  $\bar{\mathbf{T}}^{\text{ip}}$  and bending moment  $\bar{\mathbf{M}}^{\text{ip}}$  are then found by taking the partial derivatives of  $\bar{\Phi}$ :

$$\bar{\mathbf{T}}^{\text{ip}} = \frac{1}{J} \bar{\mathbf{F}}^{\text{ip}} \frac{\partial \bar{\Phi}}{\partial \bar{\mathbf{E}}^{\text{ip}}} \bar{\mathbf{F}}^{\text{ip}T} \quad \text{and} \quad \bar{\mathbf{M}}^{\text{ip}} = \frac{1}{J} \frac{\partial \bar{\Phi}}{\partial \bar{\mathbf{C}}^{\text{ip}}}, \tag{15}$$

where  $J$  is the local area dilatation of the surface and  $\Gamma_0$  is the surface  $\Gamma$  in its reference configuration. The resulting force of the interface is then found to be [35]:

$$\bar{\mathbf{f}} = \underbrace{\left(-\bar{T}_{\alpha\beta|\beta}^{\text{ip}} \mathbf{a}^\alpha - \bar{T}_{\alpha\beta}^{\text{ip}} b_{\alpha\beta} \bar{\mathbf{n}}\right)}_{\text{in-plane stress}} + \underbrace{\left(\bar{M}_{\alpha\beta|\alpha\beta}^{\text{ip}} \bar{v}^\perp - \bar{M}_{\alpha\beta}^{\text{ip}} b_\alpha^\lambda b_{\lambda\beta}\right)}_{\text{bending moment}} \bar{\mathbf{n}}. \tag{16}$$

Here, the vector  $\bar{\mathbf{f}}$  is the force resulting from stretching and bending the interface.

### 2.5. Weak formulation

Introducing the test functions  $\mathbf{w}_v, w_p, w_{\bar{v}^\parallel}, w_{\lambda_p}, w_{\lambda^+},$  and  $w_{\lambda^-}$ , integrating by parts and using the divergence theorem, the weak form of the governing Equations (7)–(10) in the fluid domain can be written as given the position  $\mathbf{r}$  of the interface  $\Gamma$  at time  $t$ , find  $\mathbf{v} \in \mathcal{V}, p \in \mathcal{P}, \bar{v}^\parallel \in \mathcal{L}, \lambda_p \in \mathcal{L}, \lambda^+ \in \mathcal{L}$  and  $\lambda^- \in \mathcal{L}$ , such that for all  $\mathbf{w}_v \in \mathcal{V}, w_p \in \mathcal{P}, w_{\bar{v}^\parallel} \in \mathcal{L}, w_{\lambda_p} \in \mathcal{L}, w_{\lambda^+} \in \mathcal{L}$  and  $w_{\lambda^-} \in \mathcal{L}$ :

$$\begin{aligned} (\nabla \mathbf{w}_v, \mu \nabla \mathbf{v})_\Omega - (\nabla \mathbf{w}_v, p \mathbf{I})_\Omega + \left(\mathbf{w}_v, \frac{\mu}{l^+} \lambda^+ - \frac{\mu}{l^-} \lambda^-\right)_\Gamma + (\mathbf{w}_v, \bar{\mathbf{f}})_\Gamma &= 0 \\ (w_p, \nabla \cdot \mathbf{v})_\Omega + (w_p, \lambda_p)_\Gamma &= 0 \\ (w_{\lambda_p}, [p])_\Gamma + (w_{\lambda_p}, \bar{\mathbf{f}}^\perp)_\Gamma &= 0 \\ (w_{\bar{v}^\parallel}, \frac{\mu}{l^+} (\lambda^+ + [v]^{\parallel+}) + \frac{\mu}{l^-} (-\lambda^- + [v]^{\parallel-}))_\Gamma + (w_{\bar{v}^\parallel}, \bar{\mathbf{f}}^\parallel)_\Gamma &= 0 \\ (w_{\lambda^+}, ((\boldsymbol{\sigma} \cdot \bar{\mathbf{n}})^{\parallel+} - \frac{\mu}{l^+} [v]^{\parallel+}))_\Gamma &= 0 \\ (w_{\lambda^-}, ((\boldsymbol{\sigma} \cdot \bar{\mathbf{n}})^{\parallel-} - \frac{\mu}{l^-} [v]^{\parallel-}))_\Gamma &= 0. \end{aligned} \tag{17}$$

Here, we introduced the Lagrange multipliers  $\lambda_p$ , which enforce the pressure jump across the interface (9), and  $\lambda^+$  and  $\lambda^-$ , which ensure the Navier boundary conditions (10) are respected on each side of the interface.

## 3. NUMERICAL SOLUTION: EXTENDED FINITE ELEMENT AND GRID-BASED PARTICLE METHOD

The solution strategy adopted here aims to attain a state of equilibrium between the interface and the surrounding fluid by solving a sequence of steady state flow and update of the interface in a staggered manner. As we march forward in time, the interface is first considered fixed, and the X-FEM is used to solve for the Eulerian fields  $\mathbf{v}^\pm, \bar{\mathbf{v}}$ , and  $p^\pm$  when the system is subjected to the interface force  $\bar{\mathbf{f}}$ . The coupling between the interface and the surrounding fluid is handled via the use of Lagrange multipliers, where special care is taken in defining the right Lagrange multiplier space along the interface [39]. This first step is described in Section 3.1. Next, given the interface velocity field  $\bar{\mathbf{v}}$ , Section 2.5 describes how the grid-based particle method and a second-order Runge–Kutta time integration scheme are used to update the interface position and deformations measures as well as the interface force  $\bar{\mathbf{f}}$  for the next time step.

3.1. The extended finite element method with directional discontinuities

The examples chosen here to illustrate the method are axisymmetrical or plane-stress and therefore only require a two-dimensional spatial discretization. Every aspect of the method can however be adapted in three dimensions but involve additional computational aspect (such as parallel computing and optimization), which are beyond the scope of this study. Following classical methods for fluid flow at low Reynolds number, the pressure and velocity fields are discretized on a two-dimensional fixed finite element mesh and interpolated with a mixed finite element procedure [44] with (four-node) bilinear shape functions for the pressure and (nine-node) quadratic shape functions for the velocity. Because the framework adopted to describe the system is Eulerian, the finite element discretization is structured and does not conform to the shape of the membrane  $\Gamma$ . The existence of an interface is then accounted for by the zero-level of a level-set function  $\phi(\mathbf{x})$  defined as the signed distance function from the interface  $\Gamma$ . An illustration of this function is given in Figure 2(a) and (b). The presence of the interface creates discontinuities in the velocity and pressure fields within elements cut by  $\Gamma$ , which can be accounted for by adopting the X-FEM: elements cut by the interface are enriched with ‘jump’ degrees of freedom, as described in [38, 45]. For example, the general rule for interpolating the pressure  $p$  inside element  $e$  is

$$\tilde{p}^e(\mathbf{x}, t) = \sum_{I=1}^4 N_I^4(\mathbf{x}) p^I(t) + \sum_{I=1}^4 N_I^4(\mathbf{x}) (H(\phi(\mathbf{x})) - H(\phi(\mathbf{x}_I))) \hat{p}^I(t) \quad (18)$$

where the upper case indices  $I$  and  $J$  are used for node numbering and the  $N_I^4$  are the four-node element shape functions. The terms  $p^I$  denotes the regular degree of freedom for the pressure at node  $I$ , while  $\hat{p}^I$  represents the degree of freedom for the jump of pressure in elements cut by  $\Gamma$ . The Heaviside function  $H(\phi(\mathbf{x}))$  provides the discontinuity needed to describe the jump in velocity and pressure across the membrane (Figure 2(c)). In contrast to the pressure field, the discontinuity that affects the velocity field is directional because only the tangential components of  $\mathbf{v}$  are discontinuous across  $\Gamma$ . We note here that a discontinuous fluid viscosity across the interface would cause a jump in the velocity gradient in the normal direction. While this aspect does not represent a particular challenge in the numerical formulation, we choose to not consider this situation for the sake of clarity. We use modified shape functions by making the discontinuity directional as follows [40]:

$$\tilde{v}_i^e(\mathbf{x}, t) = \sum_{J=1}^9 N_J^9(\mathbf{x}) v_i^J(t) + \sum_{J=1}^9 N_J^9(\mathbf{x}) (H(\phi(\mathbf{x}, t)) - H(\phi(\mathbf{x}_J, t))) a_i^J(t) \hat{v}^J(t), \quad (19)$$

where the lower case  $i$  indicates the component of a vector and  $N^9$  are the nine-node element shape functions. The term  $a_i^J$  represents the  $i^{th}$  component of the covariant tangent vector to  $\Gamma$  at node  $J$  and provides the tangential direction for the velocity jump  $\hat{v}^J$  at node  $J$ . With this method of interpolation, the continuity condition for the normal velocity  $v^{\perp+} = v^{\perp-}$  is automatically satisfied across  $\Gamma$  and the jump in tangential velocity is reduced to a scalar  $\hat{v}^J$  multiplied by the tangent vector  $a_i^J$ . Note that in the three-dimensional case, the velocity discontinuity becomes a two-dimensional vector in the plane tangent to the interface.

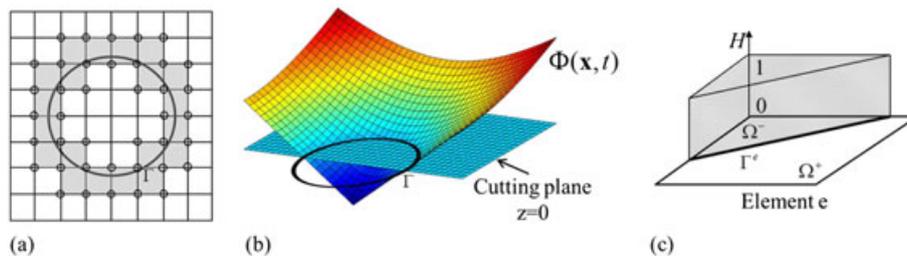


Figure 2. (a) The elements split by membrane  $\Gamma$  and the enriched nodes (circle). (b) the 3D distance function can be seen where its intersection with the plan  $z = 0$  defines  $\Gamma$ , and (c) the Heaviside function in an element cut by  $\Gamma$ .

The one-dimensional fields defined on the interface such as Lagrange multipliers  $\lambda_p, \lambda^+, \text{ and } \lambda^-$  or the membrane velocity  $\bar{v}^{\parallel}$  is interpolated along the section  $\Gamma^e$  of  $\Gamma$  that cuts element  $e$  as

$$\tilde{f}(\xi, t) = \sum_{K=1}^2 \bar{N}_K(\xi) f_K(t), \tag{20}$$

with  $\bar{f}$  a one-dimensional field defined on the interface and where  $\bar{N}_I$  are one-dimensional two-node shape functions with coordinate  $\xi$  running along  $\Gamma^e$ . Special care needs to be given to the discretization of the Lagrange multipliers along the interface in the context of the X-FEM to satisfy the inf-sup condition. The reader is referred to the work of Mões *et al* [39] for a detailed description of such discretization and its extension to three dimensions [46]. Another common issue that arises in X-FEM formulations is the typically ill-conditioned tangent matrix in (21) [47]. This is caused by the interface cutting through elements in a way that leaves very little support for the enrichment functions. This can lead to large oscillations in the pressure and velocity fields and affect the rate of convergence. Several methods have been developed to address that issue, such as using a modified step enrichment function [48] or applying X-FEM specific preconditioners [49, 50]. Here, we chose to use the blocking strategy, which consists in removing the enriched degrees of freedom that have very little support, as described in [51].

### 3.2. Discretized governing equations

After a long but straightforward calculation using the discretized form of  $\mathbf{v}, p, \bar{v}^{\parallel}, \lambda_p,$  and  $\lambda^{\pm}$  (given in the Appendix), the weak form (17) at time  $t$  can conveniently be written in the following format:

$$\begin{bmatrix} \mathbf{K}^t & \mathbf{I}_2^t \\ \mathbf{I}_1^t & \bar{\mathbf{K}}^t \end{bmatrix} \cdot \begin{bmatrix} \mathbf{d}(t) \\ \bar{\mathbf{d}}(t) \end{bmatrix} + \begin{bmatrix} \mathbf{F}_f^t \\ \bar{\mathbf{F}}_f^t \end{bmatrix} = 0 \tag{21}$$

where  $\mathbf{d}$  is a vector composed of the bulk degrees of freedom  $\mathbf{d}(t) = [\mathbf{v}(t) \mathbf{p}(t)]^T$ , while  $\bar{\mathbf{d}}$  corresponds to the interface degrees of freedom  $\bar{\mathbf{d}}(t) = [\bar{v}^{\parallel}(t) \lambda_p(t) \lambda^+(t) \lambda^-(t)]^T$ . The submatrix  $\mathbf{K}^t$  corresponds to the fluid domain and  $\bar{\mathbf{K}}^t$  is associated with the interface, while  $\mathbf{I}_1^t$  and  $\mathbf{I}_2^t$  can be thought of as the fluid/interface interaction matrices. The global force vector  $[\mathbf{F}_f^t \ \bar{\mathbf{F}}_f^t]^T$ , which include both external forces and the interface force at time  $t$ , as well as the tangent matrix components  $\mathbf{K}^t, \bar{\mathbf{K}}^t, \mathbf{I}_1^t$  and  $\mathbf{I}_2^t$  are assembled from their element constituents as follows:

$$\begin{aligned} \mathbf{F}_f^t &= \mathcal{A}_{e=1}^n (\mathbf{F}_{fe}^t) \ , \ \bar{\mathbf{F}}_f^t = \mathcal{A}_{e=1}^m (\bar{\mathbf{F}}_{fe}^t) \ , \ \mathbf{K}^t = \mathcal{A}_e^n (\mathbf{K}_e^t) \ , \ \bar{\mathbf{K}}^t = \mathcal{A}_e^m (\bar{\mathbf{K}}_e^t) \ , \\ \mathbf{I}_1^t &= \mathcal{A}_e^n (\mathbf{I}_{e1}^t) \ \text{and} \ \mathbf{I}_2^t = \mathcal{A}_e^m (\mathbf{I}_{e2}^t) \end{aligned}$$

where  $\mathcal{A}_e, n$  and  $m$  respectively denote the conventional FEM assembly operator, the total element number, and the number of elements cut by  $\Gamma$ . Precise expressions of the submatrix and subvectors appearing in (21) are provided in the Appendix.

The finite element Equation (21) can be solved with a linear solver and yield the fluid and interface velocities at time  $t$ . Given the interface velocity  $\bar{\mathbf{v}}(t)$ , the position of  $\Gamma$  and its deformation measures  $\bar{\mathbf{C}}^t$  and  $\bar{\mathbf{F}}^t$  are then updated using an explicit second-order Runge–Kutta time integration scheme, as discussed in the next section.

### 3.3. Tracking the evolution of the interface

Traditionally, the temporal evolution of an interface in an Eulerian framework is handled with the level-set evolution equation [25]. This has the main advantage of turning the evolution of the interface into seeking the solution of a partial differential equation, removing the need for surface parameterization or meshing. The main issue with this class of methods within the context of our study is that a level-set formulation is usually not adapted to describe Lagrangian information on an interface, which may include strain and concentration fields. They also suffer from the fact that they generally cannot capture in-plane, interface shear deformation because such instances are not associated with a normal interface velocity (the latter being the main component of the level-set evolution

equation). To circumvent these limitations, we choose here to enrich the level-set with particles by using a grid-based particle method similar to what was introduced in [41]. This method possesses the double advantage of tracking the interface explicitly with particles that contain Lagrangian information while using the underlying fixed finite element mesh to ensure a fairly uniform repartition of the particles on the interface. Although particle-enriched level-set methods have been developed to study two phase flows [52], they have not, to the knowledge of the authors, been applied to the modeling of immersed elastic membranes and therefore were not used to carry Lagrangian information on the interface. Here, we summarize the grid-based particle method and discuss the update of the interface position and deformations measures.

The particles, whose position vector is denoted by  $\mathbf{y}$ , are chosen as the normal projection of the underlying mesh nodes, with position vector  $\mathbf{p}$  on  $\Gamma$ . Initially, the interface is described implicitly as the zero level-set of a signed distance function  $\phi(\mathbf{p}, t = 0)$ . The initial coordinates of particles  $\mathbf{y}$  can then be found as follows:

$$\mathbf{y} = \mathbf{p} - \phi(\mathbf{p}, 0)\nabla\phi(\mathbf{p}, 0). \tag{22}$$

To limit the number of particles, we define a so-called computational tube such that only nodes  $\mathbf{p}$  whose distance to  $\Gamma$  is smaller than a cut-off value  $\lambda_{tube}$  are accounted for (Figure 3(a)). It is important to note here that there is a one-to-one correspondence between each particle  $\mathbf{y}$  and node  $\mathbf{p}$ . This ensures a quasi-uniform repartition of particles along the interface throughout its evolution. Between two subsequent time steps, the particles are moved according to the interface velocity  $\tilde{\mathbf{v}}(\xi, t)$  and using a second-order Runge–Kutta time-integration procedure as follows (Figure 3(b)):

$$\mathbf{y}^{t+dt/2} = \mathbf{y}^t + \tilde{\mathbf{v}}(\mathbf{y}^t, t)\frac{dt}{2} + \mathbf{\Omega} \cdot \tilde{\mathbf{v}}(\mathbf{y}^t, t)\frac{dt^2}{4} \tag{23}$$

$$\mathbf{y}^{t+dt} = \mathbf{y}^t + \tilde{\mathbf{v}}(\mathbf{y}^{t+dt/2}, t)dt + \mathbf{\Omega} \cdot \tilde{\mathbf{v}}(\mathbf{y}^{t+dt/2}, t)\frac{dt^2}{2}, \tag{24}$$

where  $\mathbf{\Omega}$  is the matricial form of the angular velocity of the interface normal [53]:

$$\frac{d\bar{\mathbf{n}}}{dt} = \boldsymbol{\omega} \times \bar{\mathbf{n}} \quad , \quad \boldsymbol{\omega} = -\left(v_{,\alpha}^\perp + b_{\lambda\alpha}v^{\parallel\lambda}\right)\mathbf{a}^\alpha \quad \text{and} \quad \Omega_{ik} = \epsilon_{ijk}\omega_j \tag{25}$$

with  $\epsilon_{ijk}$  the permutation tensor. After the motion of the interface, the particles  $\mathbf{y}$  may not be the closest points on  $\Gamma$  to their associated nodes  $\mathbf{p}$ . Moreover, the motion of the particles may cause their distribution on  $\Gamma$  to become uneven, which can affect the geometrical resolution of the interface. To overcome this issue, the interface is resampled after motion by recomputing the position of particles as the closest points on  $\Gamma$  to the nodes  $\mathbf{p}$  inside the updated computational tube (which has moved with the interface). This is carried out by first locally approximating the interface with polynomials

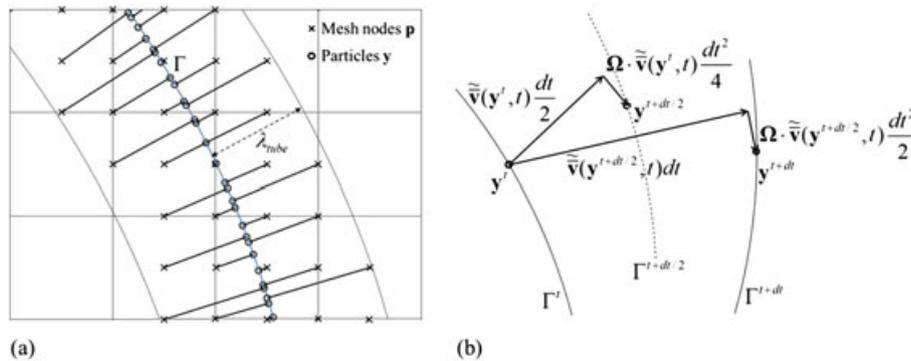


Figure 3. (a) The sampling of  $\Gamma$  by the particles  $\mathbf{y}$  represented by a circle, while the associated nodes  $\mathbf{p}$  inside the computational tube  $\lambda_{tube}$  are represented by crosses and (b) the second-order Runge–Kutta time integration of the evolution of the interface:  $\tilde{\mathbf{v}}$  denotes the velocity of the interface interpolated at the particle’s position, while  $\mathbf{\Omega}$  is the angular velocity of the interface’s normal  $\bar{\mathbf{n}}$ .

around each particle. The procedure, explained here in the two-dimensional case, is as follows: for each node  $\mathbf{p}$  inside the computational tube, the closest  $m$  particles  $\mathbf{y}_0 \dots \mathbf{y}_m$  are collected, carrying with them the tangent  $\mathbf{a}'_0 \dots \mathbf{a}'_m$  and normal  $\bar{\mathbf{n}}'_0 \dots \bar{\mathbf{n}}'_m$  to the interface before motion. Denoting  $\mathbf{y}_0$  as the particle closest to  $\mathbf{p}$ , a polynomial of degree  $n < m$  is fitted to the particles  $\mathbf{y}_0 \dots \mathbf{y}_m$  in the local coordinate system  $\{\mathbf{a}'_0; \bar{\mathbf{n}}'_0\}$  centered on  $\mathbf{y}_0$ . The location  $\tilde{\mathbf{y}}_i$  of particle  $i$  in this local coordinate system is given by

$$\tilde{\mathbf{y}}_i = \begin{Bmatrix} \xi_i^1 \\ \xi_i^2 \end{Bmatrix} = \mathbf{R}^t \cdot (\mathbf{y}_i - \mathbf{y}_0) \quad \text{with} \quad \mathbf{R}^t = \begin{bmatrix} (\mathbf{a}'_0)^T \\ (\bar{\mathbf{n}}'_0)^T \end{bmatrix}. \tag{26}$$

Taking the example of a quadratic polynomial ( $n = 2$ ), the interface around particle  $\mathbf{y}_0$  is represented in the local referential as the graph function  $\xi^2(\xi^1) = c_0 + c_1\xi^1 + c_2(\xi^1)^2$ , where the coefficients  $c_0, c_1$ , and  $c_2$  are found by minimizing the  $L^2$  difference between the  $\xi^2(\xi^1)$  and the  $\xi_i^2$ . This method of interface approximation can easily be generalized in three dimension [41]. The coordinates  $\{\xi^1, \xi^2(\xi^1)\}$  defines a local parameterization  $\mathbf{r}^l(\xi^1)$  of  $\Gamma$  in the neighborhood of  $\mathbf{y}_0$ :

$$\mathbf{r}^l(\xi^1) = \begin{Bmatrix} \xi^1 \\ \xi^2(\xi^1) \end{Bmatrix}. \tag{27}$$

The relationship between the local parameterization  $\mathbf{r}^l(\xi^1)$  and the new updated global parameterization  $\mathbf{r}(\xi^1, t + dt)$  is then found via rotation and translation operations:

$$\mathbf{r}(\xi^1, t + dt) = (\mathbf{R}^t)^{-1} \mathbf{r}^l(\xi^1) + \mathbf{y}_0. \tag{28}$$

The parameterization  $\mathbf{r}(\xi^1, t + dt)$  can now be used to resample the interface, i.e. recalculate the closest point on the interface to the nodes  $\mathbf{p}$ . This is carried out by minimizing the distance function  $d(\mathbf{r}(\xi^1, t + dt), \mathbf{p})$  with respect to  $\xi^1$ . In three dimensions, this requires the use of an iterative solver, which for a small  $dt$  and given a good initial guess, converges very quickly. In two dimensions, however, the solution can be found explicitly by solving a cubic equation. The quality of approximation of a two-dimensional interface when fitted to a circle using the sequence of second-order polynomials detailed earlier is shown in Table I. It is found that the  $L^2$  error  $e^2$  in the position of the particles decreases with the size of the underlying mesh at with a convergence rate of approximately 4. This is expected because both the circle and the polynomial are even functions and third-order terms therefore do not contribute to the reconstruction error. The updated tangent vector  $\mathbf{a}_1^{t+dt}$  is found using the parameterization  $\mathbf{r}(\xi^1, t + dt)$  as

$$\mathbf{a}_1^{t+dt} = \frac{\partial \mathbf{r}(\xi^1, t + dt)}{\partial \xi^1} \mathbf{r}(\xi^1) = \mathbf{R}^t \frac{\partial \mathbf{r}^l(\xi^1)}{\partial \xi^1}, \tag{29}$$

while the normal vector  $\bar{\mathbf{n}}$  is found as the vector orthogonal to both  $\mathbf{a}_1$  and the out-of-plane direction.

Finally, the new level-set function  $\phi(\mathbf{p}, t + dt)$  is calculated as the signed distance function to  $\Gamma$ , at nodes  $\mathbf{p}$ :

$$\phi(\mathbf{p}, t + dt) = -\text{sgn} \left( \frac{\mathbf{y}^{t+dt} - \mathbf{p}}{|\mathbf{y}^{t+dt} - \mathbf{p}|} \cdot \bar{\mathbf{n}}'_0 \right) |\mathbf{y}^{t+dt} - \mathbf{p}|, \tag{30}$$

where  $\mathbf{y}^{t+dt}$  is the particle associated with  $\mathbf{p}$  at time  $t + dt$  and the term  $\text{sgn}((\mathbf{y}^{t+dt} - \mathbf{p})/|\mathbf{y}^{t+dt} - \mathbf{p}|) \cdot \bar{\mathbf{n}}'_0$  determines whether node  $\mathbf{p}$  is in  $\Omega^+$  or  $\Omega^-$ . The reconstruction of the level-set function using the local polynomial approximation of the interface in the aforementioned equation is computationally inexpensive and used the X-FEM part of the algorithm (18) and (19).

Table I.  $L^2$  error in the position of particles when fitted to a circle.

Mesh size h	0.4	0.2	0.1	0.05
$e^2$	$2.87 \times 10^{-2}$	$1.73 \times 10^{-3}$	$1.12 \times 10^{-4}$	$6.97 \times 10^{-6}$

### 3.4. Update of the interface deformation

In addition to the interface geometry, the sampling particles also carry Lagrangian fields associated with the interface, such as the deformation gradient. The in-plane deformation gradient  $\bar{\mathbf{F}}^{\text{ip}}$  is updated at each particle  $\mathbf{y}^t$  in their respective basis  $\{\mathbf{a}_1^t, \bar{\mathbf{n}}^t\}$  using the in-plane velocity gradient  $\bar{\mathbf{L}}^{\text{ip}}$  as follows [54]:

$$\bar{L}_{\alpha\beta}^{\text{ip}} = \tilde{v}_{\alpha|\beta}^{\parallel} - b_{\alpha\beta} \tilde{v}^{\perp} \tag{31}$$

$$\bar{\mathbf{F}}^{\text{ip}}(t + dt) = \bar{\mathbf{F}}^{\text{ip}}(t) \exp(\bar{\mathbf{L}}^{\text{ip}} dt), \tag{32}$$

where  $\tilde{v}^{\parallel}$  and  $\tilde{v}^{\perp}$  are the tangent and normal component of the interface velocity interpolated at particle  $\mathbf{y}^t$  and  $|\cdot|$  denotes the covariant derivative. Similarly to the polynomials that approximates the geometry of the interface in Section 3.2, a new set of polynomials is constructed to interpolate the deformation gradient around each particle. The second-order polynomial that approximates the deformation gradient  $\bar{\mathbf{F}}^{\text{ip}}(t + dt)$  around a particle  $\mathbf{y}^{t+dt}$  is written as

$$\bar{F}_{\alpha\beta}^{\text{ip}}(\xi_1, t + dt) = c_0^{\alpha\beta} + c_1^{\alpha\beta} \xi_1 + c_2^{\alpha\beta} (\xi_1)^2, \tag{33}$$

where the coefficients  $c_i^{\alpha\beta}$  are found with the method of least square fitting, using the values of the deformation gradient at the neighboring particle. Thanks to the local polynomial reconstructions, the deformation gradient  $\bar{\mathbf{F}}^{\text{ip}}(t + dt)$  can be interpolated anywhere on the interface  $\Gamma$ . Once the different deformation measures of the interface are updated, one can compute the interface force  $\bar{\mathbf{f}}$  given by Equation (16). The algorithm then marches forward in time and computes the velocity and pressure fields for the new time step using Equation (21).

### 3.5. Convergence criteria

Convergence is achieved once the two following criteria are met:

- (1) The pressure and velocity fields of the fluid surrounding the interface are constant with respect to time. This criterion is met when  $\|\mathbf{v}_{\text{reg}}(t + dt) - \mathbf{v}_{\text{reg}}(t)\| < \text{tol}_v$  and  $\|\mathbf{p}_{\text{reg}}(t + dt) - \mathbf{p}_{\text{reg}}(t)\| < \text{tol}_p$ , where  $\mathbf{v}_{\text{reg}}(t)$  and  $\mathbf{p}_{\text{reg}}(t)$  are vectors that contain all regular fluid pressure and velocity degrees of freedom at each node and  $\text{tol}_v$  and  $\text{tol}_p$  the user-defined tolerances for the velocity and pressure.
- (2) The interface tangential velocity field is constant with respect to time. Here, because the number of degrees of freedom associated with the interface changes at each iteration, we rely on the particles tracking the interface (described in the next section) to compare the interface velocity field between two time steps. The second criterion is therefore written  $\|\bar{\mathbf{v}}_p^{\parallel}(t + dt) - \bar{\mathbf{v}}_p^{\parallel}(t)\| < \text{tol}_{\bar{v}^{\parallel}}$ , with  $\bar{\mathbf{v}}_p^{\parallel}(t + dt)$  and  $\bar{\mathbf{v}}_p^{\parallel}(t)$  the tangential velocity of the interface interpolated at the same particles at times  $t$  and  $t + dt$ .

Once criteria (1) and (2) are satisfied, the algorithm has converged and the interface is in equilibrium with the surrounding fluid.

## 4. NUMERICAL EXAMPLES

We now consider several numerical examples with the aim of validating the proposed model and illustrating its ability to simulate some cases of extreme membrane deformations occurring in engineering and biological problems.

### 4.1. Convergence study

Here, we investigate the convergence of the error made in calculating the flow velocity and pressure around a fixed, rigid spherical capsule. We then allow the capsule to elastically deform in contact with the surrounding fluid and study the CFL condition in time step  $dt$  for the temporal evolution of the interface.

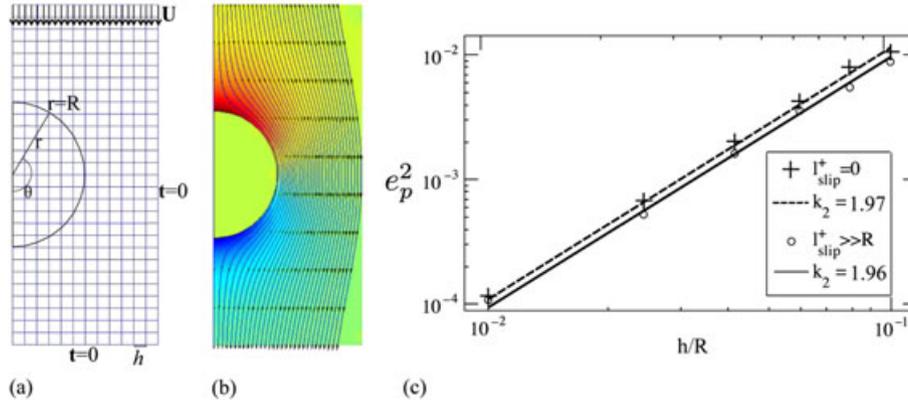


Figure 4. (a) Finite element mesh and boundary conditions, (b) pressure field and velocity streamline for the fluid flowing around the sphere, and (c) convergence of the error made on the pressure on the surface of the sphere as the mesh size decreases.

4.1.1. *Rigid spherical capsule.* Let us consider a uniform incompressible flow past a rigid spherical capsule and compare the model’s prediction and the analytical solution for the flow velocity and pressure around the sphere with ‘slip’ or ‘stick’ boundary conditions at its surface. Analytically, the solution of the Stokes flow for an infinite expanse of fluid around a rigid sphere is given, in spherical coordinates (Figure 4(a)), by [55]:

$$v_r = \frac{1}{r^2 \sin^2(\theta)} \frac{\partial \Psi}{\partial \theta}, \quad v_\theta = \frac{-1}{r \sin \theta} \frac{\partial \Psi}{\partial r}, \quad v_\phi = 0 \tag{34}$$

where  $\Psi$  is the Stokes stream function that takes the form in the ‘stick’ and ‘slip’ cases, respectively [55]:

$$\Psi^{\text{stick}}(r, \theta) = \sin^2(\theta) \frac{U}{4} \left( \frac{R^2}{r} - 3Rr + 2r^2 \right) \tag{35}$$

$$\Psi^{\text{slip}}(r, \theta) = \sin^2(\theta) \frac{U}{2} (-Rr + r^2). \tag{36}$$

with  $R$  is the radius of the sphere and  $U$  is the flow velocity away from the sphere  $v_{r \rightarrow \infty} = U$ . In our model, the ‘stick’ condition corresponds to a Navier boundary condition with  $l_{slip}^+ = 0$ , while the ‘slip’ boundary condition requires  $l_{slip}^+ \gg R$ . We first investigate the convergence of the error made in calculating the pressure at the surface of the rigid sphere as the mesh size  $h$  is decreased. We define the  $L^2$ -error in pressure as

$$e_p^2 = \frac{1}{\pi p_{\max}} \left[ \int_0^\pi (p_{\text{exact}}(r = R, \theta) - p_{\text{num}}(r = R, \theta))^2 d\theta \right]^{1/2} \tag{37}$$

where  $p_{\text{exact}}$  can easily be obtained from the Stokes stream functions (35) and (36),  $p_{\max} = \max(p_{\text{exact}}(r = R, \theta))$  and  $p_{\text{num}}$  is the pressure calculated numerically. Figure 4(b) presents the pressure field around the sphere for the ‘stick’ case, while Figure 4(c) shows that the error made in calculating the pressure converges to zero like  $O((h/R)^3)$ , for both extreme cases  $l^+ = 0$  (stick) and  $l^+ \gg R$  (slip). The Reynolds number chosen in those simulations is  $Re = RU\rho/(\mu) = 0.05$ , at which inertia forces can be neglected. The solid and dotted lines show the power law fitting of the error, calculated as

$$e_p^{\text{fit}} = k_1 (h/R)^{k_2} \tag{38}$$

where  $k_2 \sim 2$  is an approximation of the rate of convergence for the error in pressure in both ‘slip’ and ‘stick’ cases. This is consistent with the theoretical convergence rates of a bilinear/biquadratic approximation for the pressure/velocity fields [56].

Figure 5(a) and (b) shows the vertical flow velocity  $\mathbf{v}_\theta(r, \theta = \pi/2)$  and the surface pressure  $p(r = R, \theta)$  for a rigid sphere with different slip length, ranging from  $l^+ = 0$  (‘stick’ condition) to  $l^+ \gg R$  (‘slip’ condition). We observe an excellent agreement between the analytical and numerical results for the vertical flow velocity  $\mathbf{v}_\theta(r, \theta = \pi/2)$  and surface pressure  $p(r = R, \theta)$  in both the ‘stick’ and ‘slip’ cases. Intermediate values of  $l^+$  show the capability of the model to continuously transition from a ‘stick’ to ‘slip’ boundary condition. As we increase the slip length from  $l^+ = 0$ , the tangential velocity jump increases (Figure 5(a)), while the viscous shear stress at the surface of the sphere decreases because  $(\boldsymbol{\sigma} \cdot \bar{\mathbf{n}})^{\parallel+} = (\mu[\mathbf{v}]^+)/l^+$ , which in turn lowers the pressure exerted by the fluid on the surface.

*Remark*

In the enrichment functions that allow the tangential velocity jump for  $l \gg R$ , we use the tangent vectors calculated from the second-order polynomials that approximate the interface. However, the Gaussian quadrature used to calculate the stiffness matrix in split elements uses a subset of triangles, which leads to a piece-wise linear interface. This inconsistency is believed to lead to the suboptimal convergence rates [51] observed in Figure 6(a). A possible solution would consist in adopting a high-order X-FEM [57] to match the order of approximation of the polynomials.

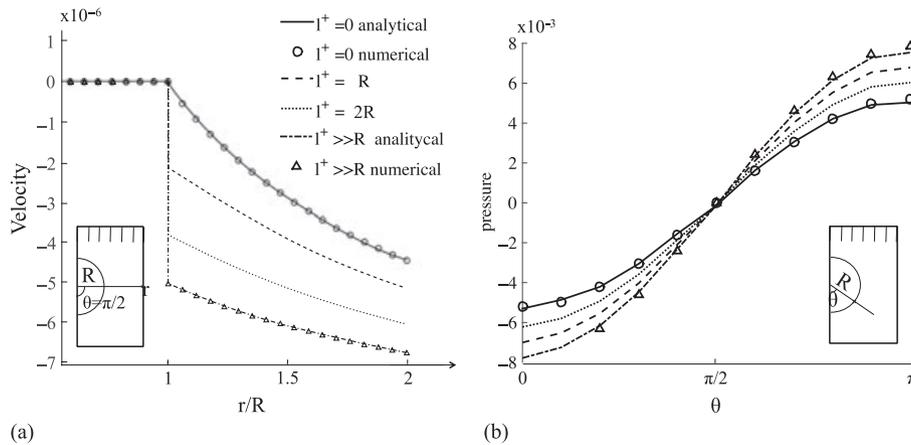


Figure 5. Simulation and analytical results of the (a) velocity and (b) pressure for different slip lengths. For  $l^+ > 0$ , there exists a jump in the tangential fluid velocity at the surface of the sphere ( $r/R=1$ ). This jump in tangential velocity is shown to decrease to 0 with  $l^+$ .

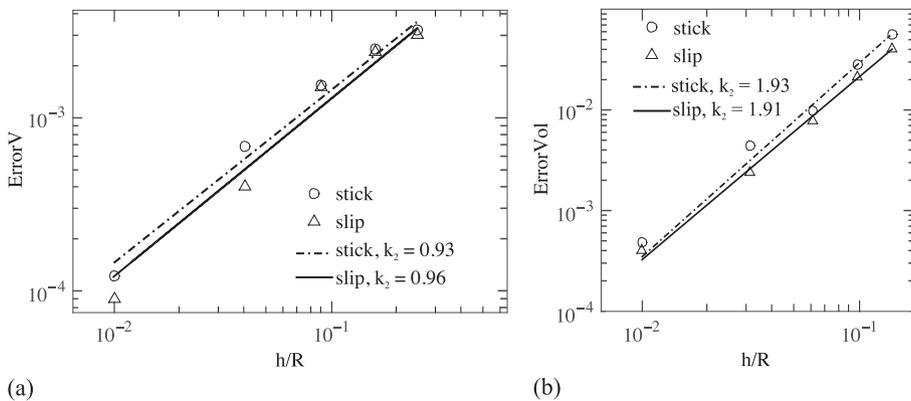


Figure 6. Convergence of the error made in conserving the (a) enclosed volume and (b) residual velocity as the mesh size decreases, for  $\Delta t \sim O(h/R)^4$  and for different slip lengths (symbols), and their corresponding  $e_p^{\text{fit}}$  (lines).

4.1.2. *Soft spherical capsule.* Let us now include the evolution of the interface by allowing the elastic spherical capsule to deform in contact with the surrounding fluid, while its top extremity remains fixed. We first endow the capsule with elastic mechanical properties. To this end, we write the capsule’s strain energy function  $\bar{\phi}(\bar{\mathbf{E}}^{\text{ip}}, \bar{\mathbf{C}}^{\text{ip}})$  as the sum of a quadratic potential function of  $\bar{\mathbf{E}}^{\text{ip}}$  and the Helfrich surface energy function of the mean curvature  $H$  [34]:

$$\bar{\phi}(\bar{\mathbf{E}}^{\text{ip}}, \bar{\mathbf{C}}^{\text{ip}}) = \frac{1}{2} \bar{\mathbf{E}}^{\text{ip}} : \mathbf{C} : \bar{\mathbf{E}}^{\text{ip}} + \frac{\kappa}{2} (H - H_0)^2, \tag{39}$$

where the spontaneous curvature was taken to be  $H_0 = 0$ . Next, to characterize the capsule’s elasticity relative to the fluid velocity, we introduce a dimensionless parameter  $G = (\mu U)/E$ , where  $E = \bar{\mu}(3\bar{\lambda} + 2\bar{\mu})/(\bar{\lambda} + \bar{\mu})$  is the in-plane elastic modulus of the capsule. The membrane Poisson’s ratio was taken to be  $\nu = \bar{\lambda}/(2(\bar{\lambda} + \bar{\mu})) = 0.25$ . A second parameter  $E_b = \kappa/(R^2 E)$  relates the bending elasticity  $\kappa$  to  $E$ .

Figure 6 plots the convergence of the residual velocity of the membrane as well as the error made in conserving the enclosed volume for values of  $G$  and  $E_b$  that produce large deformations of the capsule’s membrane. For both cases  $l^+ = 0$  and  $l^+ \gg R$ , the convergence rate for the velocity is close to  $O(h/R)$  (Figure 6(a)), which is consistent with the theoretical rates for a biquadratic/bilinear approximation of the velocity and pressure. Furthermore, the error made in conserving the enclosed volume converges like  $O(h/R)^2$  (Figure 6(b)) because of the second-order polynomials approximation of the interface geometry and the second-order Runge–Kutta algorithm used for the update of the position of the interface. However, because of the presence of fourth-order terms in the membrane bending force, the explicit time evolutive simulations are subjected to a strict Courant-Friedrichs-Lewy (CFL) condition on the time step of the fourth order in mesh size  $\Delta t \sim O(h/R)^4$  [58].

The shapes adopted at equilibrium by the capsule with  $l^+ = 0$  and for different values of  $G$  are shown in Figure 7(a). As expected, one can see that the capsule deformation increases with the ratio of the fluid velocity to capsule elasticity. Figure 7(b) illustrates the effect of the slip length  $l^+$  on the capsule deformation for  $G = 0.1$ . As the slip length increases, the pressure exerted on the capsule

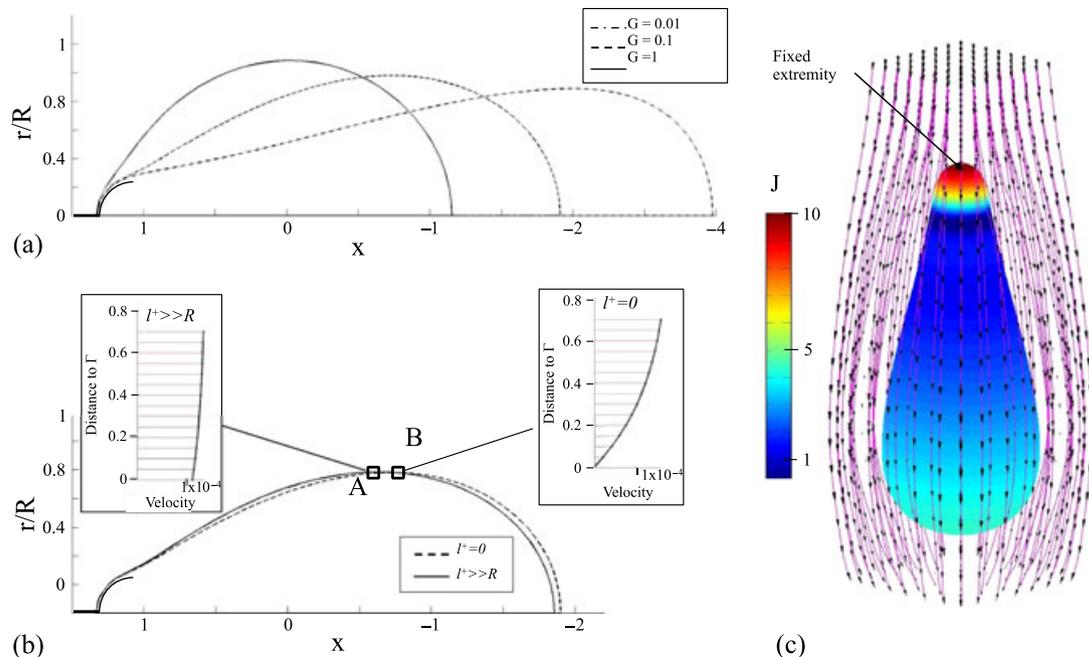


Figure 7. (a) Capsule deformations for different values of  $G$  and (b) effect of the slip length on the capsule deformation and velocity profile. (c) fluid streamlines around the capsule and the surface Jacobian  $J$ . The time step used was  $\Delta t = (h/R) \times 1e^{-4}$  and the tolerances  $\text{tol}_v = \text{tol}_p = \text{tol}_{\bar{v}} = 1e^{-5}$ .

decreases, which in turn reduces the deformation. Furthermore, Figure 7(b) shows the fluid velocity profiles along a direction normal to the capsule at points A and B, respectively, for  $l^+ \gg R$  and  $l^+ = 0$ . The discontinuity in tangential velocity between the fluid and the membrane is clearly apparent in the case  $l^+ \gg R$ , while setting  $l^+ = 0$  ensures a continuous tangential velocity that decreases to zero at the surface of the capsule. As the capsule deforms under the action of the fluid flow, the Jacobian  $J$  of the deformation represented as a color gradient on the surface in Figure 7(c) is shown to take very large values around the fixed extremity. The particle resampling ensures that the spatial discretization and the geometrical resolution of the membrane remain uniform throughout the simulation.

4.2. Capsule in shear flow

Our second example investigates the equilibrium shapes of an initially circular capsule in shear flow, with the objectives of replicating the results from that in [42] for validation and showing the effect of slip length on the deformation and ‘tank-threading’ motion. The capsule is endowed with the same elastic properties as in (39), and its initial circular shape corresponds to an in-plane stress-free configuration. However, because the spontaneous curvature  $H_0$  is chosen to be zero (which corresponds to a flat surface), the capsule has a non-zero bending energy at time  $t = 0$ . The capsule is subjected to a shear flow on the top and bottom boundaries of the computational domain of size  $L_x$  by  $L_y$ , as shown in Figure 8. The shear flow is characterized by a shear rate  $k_{\text{shear}}$  with  $\mathbf{v} = (k_{\text{shear}}y, 0)$ . The magnitude of the shear flow relative to the capsule’s elasticity is captured by the dimensionless parameter  $G$  such that  $G = (\mu k_{\text{shear}}R)/E$ . The ratio between the bending elasticity  $\kappa$  and the in-plane elasticity  $E$  constitutes a second dimensionless parameter written as  $E_b = \kappa/(R^2 E)$ . The Reynolds number in these simulations is calculated as  $Re = (4R^2 k_{\text{shear}}\rho)/(\mu)$  and is set to 0.05, where inertia effects can be neglected. The size of the computational domain is  $L_x = 4R$  by  $L_y = 4R$  and is discretized using square elements of size  $h = 0.1R$ , as this was shown to be sufficient to make the simulation independent of the mesh size.

Figure 9(a) and (b) shows the streamlines, velocity magnitude, and pressure field for the ‘stick’  $l^+ = 0$  and ‘slip’  $l^+ \gg R$  capsules at steady state. As expected, the effect of the capsule’s surface property is quite apparent: for the same values of  $G$  and  $E_b$ , the ‘stick’ capsule clearly displays a

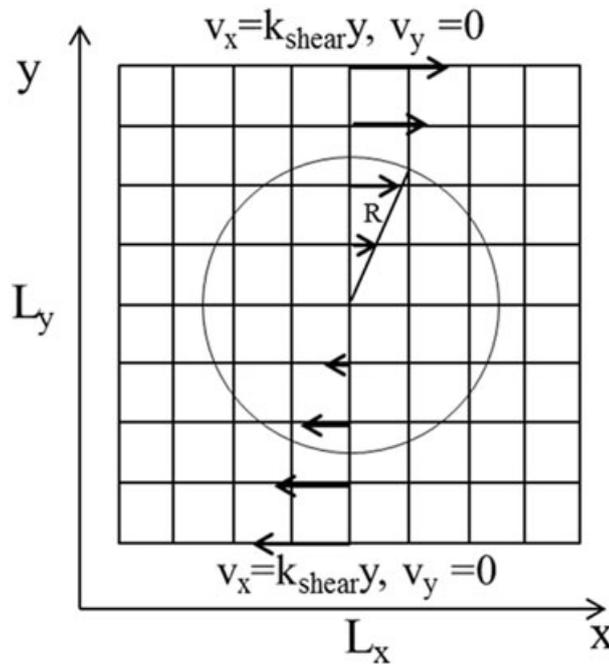


Figure 8. (a) The computational domain and the boundary conditions for the elastic capsule in shear flow.

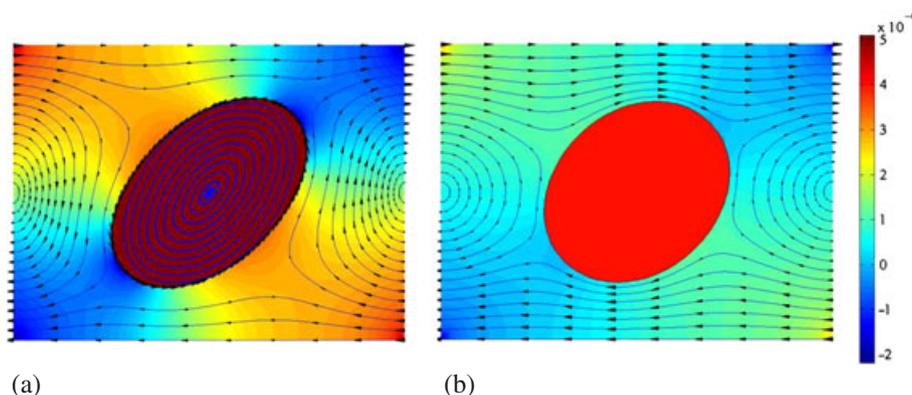


Figure 9. (a) and (b) The pressure fields, streamlines and velocity magnitude for the steady states of a capsule in shear flow, with stick and slip surfaces. The time step used was  $\Delta t = (h/R) \times 1e^{-4}$  and the tolerances  $\text{tol}_v = \text{tol}_p = \text{tol}_{\bar{v}\parallel} = 1e^{-5}$ .

tank-threading motion, as observed in [42]. However, the tank-threading motion is totally absent for a ‘slip’ capsule. The magnitude of the pressure field and the capsule deformation are also much higher in the ‘stick’ than in the ‘slip’ case because a greater slip length increases the viscous shear stress around the capsule.

The equilibrium shapes of the capsule with different values of  $E_b$  are shown in Figure 10 in the ‘slip’ and ‘stick’ cases. One can see that the effect of the bending elasticity has a direct impact on the equilibrium shapes of the capsules: a higher bending rigidity reduces the curvature at both ends of the capsule, and for a high enough value of  $E_b$ , the capsule undergoes almost no deformation. Here again, the effect of surface properties can be observed as the deformations of a ‘slip’ capsule are smaller compared with the ‘stick’ case, for equal bending elasticity. More quantitatively, the temporal evolution of the Taylor deformation parameter is presented in Figure 11. The Taylor deformation parameter is defined by  $D_{xy} = (a - b)/(a + b)$ , where  $a$  and  $b$  are the length and width of the capsule. The black dots in Figure 11(a) represent data points from that in [42] for the ‘stick’ case. Although the Taylor deformation parameter appears somewhat lower for  $E_b = 0$  and  $E_b = 0.025$ , the temporal evolution of the capsule in the ‘stick’ case is in very good agreement with the results from [42]. The ‘slip’ boundary condition is shown to decrease the maximum value of the Taylor deformation parameter compared with the ‘stick’ case for the same bending rigidities  $E_b$  in Figure 11(b).

#### 4.3. Membrane folding

To illustrate the advantages of an Eulerian formulation for the membrane, we investigate the compression of an elastic membrane lying on a fluid substrate. This corresponds for example to lipid monolayers in lung surfactant [59], often subject to large mechanical stresses as they are compressed laterally. One mechanism of stress relaxation adopted by these quasi-two-dimensional monolayers is a collapse in the out-of-plane direction and folding. Although the mechanisms behind the formation of these structures have been explored recently [60–63], there is only scattered information regarding (1) the folding dynamic after the point of self-contact has been reached and (2) the effect of substrate viscosity on the folding process.

Consider a thin elastic membrane  $\Gamma$  lying on a fluid substrate  $\Omega$  delimited by  $\partial\Omega_s$  and  $\partial\Omega_b$ , respectively, for the sides and bottom (Figure 12) and subject to the acceleration of gravity. The membrane is compressed by feeding material through the sides of the computational domain along with the fluid substrate, using a velocity boundary condition at  $\partial\Omega_b$  on both membrane and fluid. This type of problem would be challenging to implement using a Lagrangian formulation, but an Eulerian formulation allows us to seemingly add new membrane material into our computational domain.

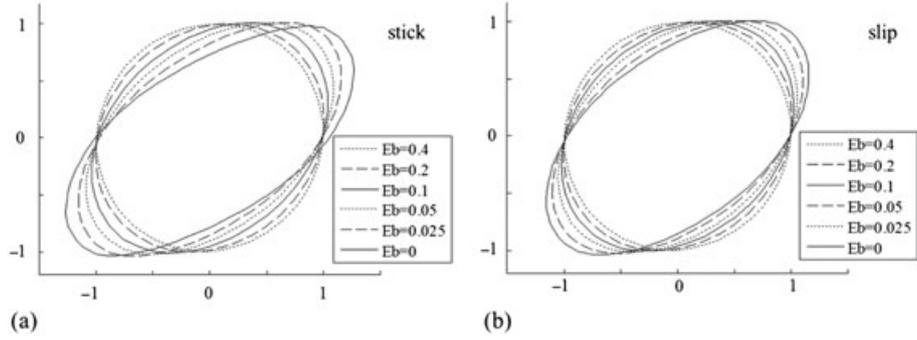


Figure 10. The capsule deformation with different bending rigidities in the cases (a)  $l^+ = 0$  (stick) and (b)  $l^+ \gg R$  (slip).

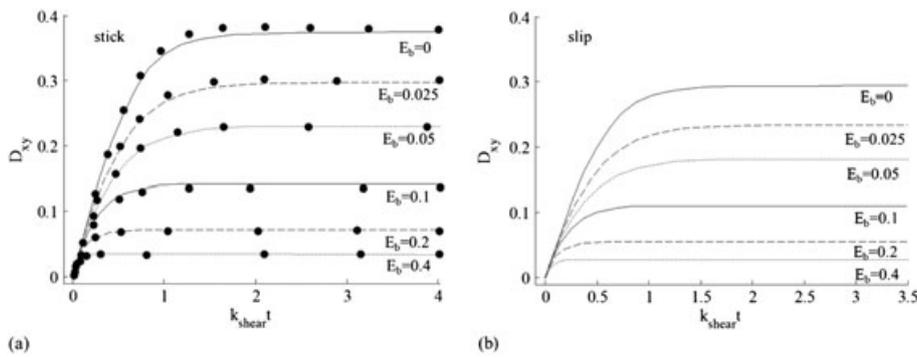


Figure 11. (a) (b) the Taylor deformation parameter  $D_{xy}$  as a function of the non-dimensional time  $k_{shear}t$  for different bending rigidities in the stick and slip cases, respectively. The black dots in (a) represent data points from that in [42].

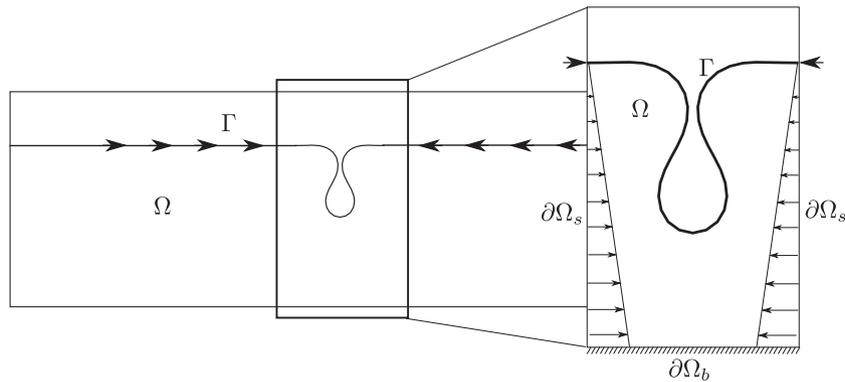


Figure 12. Compression of an elastic membrane  $\Gamma$  lying on a viscous fluid substrate  $\Omega$ . Material is fed through the domain's sides  $\partial\Omega_s$  to put the membrane under compression and initiate the folding.

The equilibrium shape of a compressed elastic membrane lying on a fluid substrate was found by Diamant and Witten [61] to have an explicit solution. The geometry of the profile is tracked by the angle  $\phi(s)$  between the membrane and the horizontal direction at arc length  $s$ . The profile of the membrane  $h(s)$  is then determined by  $\phi$  through the relation  $\dot{h} = \sin \phi$ . Further, assuming that the membrane is incompressible and remains flat and horizontal on the domains' left and right boundaries, the hydrostatic equilibrium shape of the fold is given by the following analytical expressions:

$$\phi(s) = 4 \tan^{-1} \left[ \frac{\kappa \sin(\kappa s)}{k \cosh(\kappa s)} \right] \tag{40}$$

with  $k = \frac{1}{2}(2 + P)^{1/2}$  and  $\kappa = \frac{1}{2}(2 - P)^{1/2}$ , where  $P = \text{tr}(\bar{\mathbf{T}}^{\text{ip}})$  is the membrane's surface pressure. Figure 13(a) shows snapshots of the membrane profile and the fluid pressure at different time during the folding process for very low compression rates (hydrostatic case). The analytic solution (crosses) and the results of the simulation (solid line) for the profile of the membrane under compression are shown in Figure 13(b), where we observe an excellent agreement between the two.

As we increase the compression rate, however, viscous effects start to become predominant and affect the profile of the fold. Figure 14(a) shows a snapshot of the fold profile for membranes with

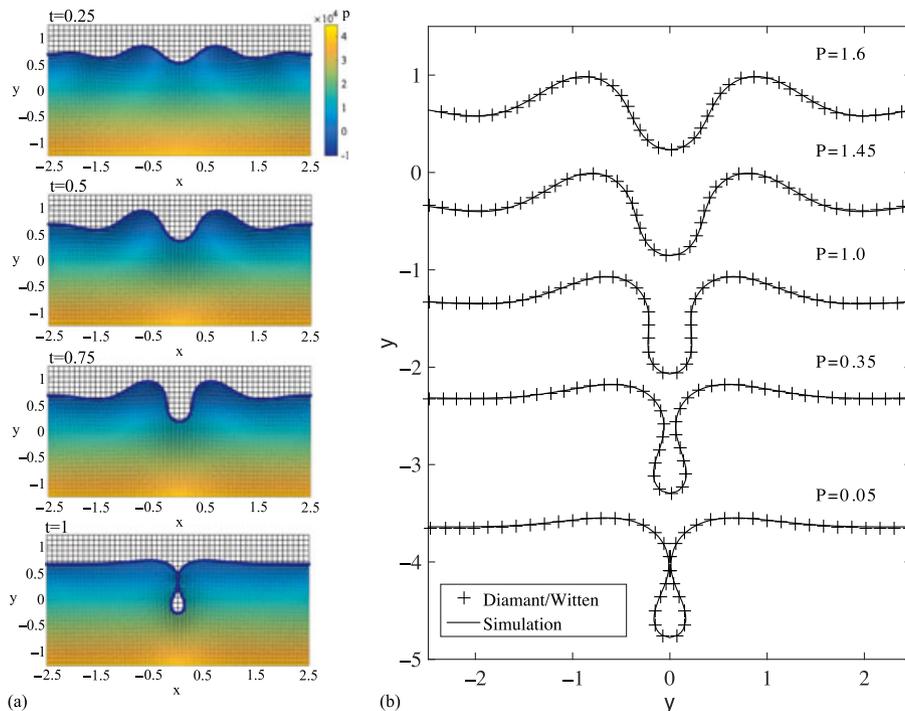


Figure 13. (a) Membrane profile and fluid pressure at different stages of the folding process, from the development of the bucking instability to the point of self-contact at  $t = 1$ . (b) Validation: membrane profile at equilibrium for different values of the membrane pressure  $P$ . The simulated deflection of the membrane at different stages (crosses) is in very good agreement with the Diamant/Witten analytic solution (solid lines).

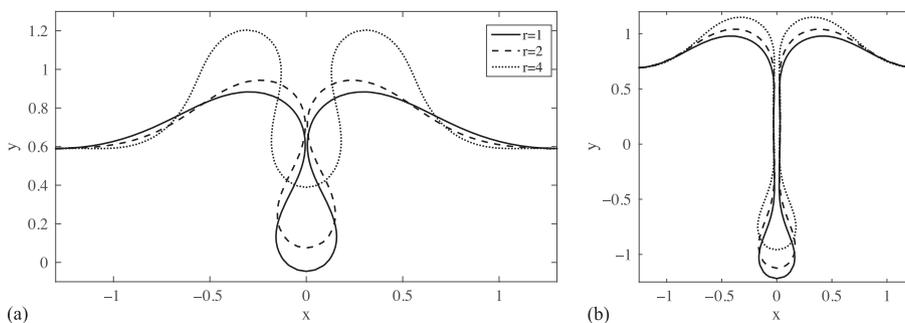


Figure 14. Effect of different compression rates  $r$  on the fold profile, for the same total membrane arc length. At high compression rates, viscous effects become dominant over hydrostatics, and prevent the fold from plunging down. As the rate decreases, the fold profile becomes that of the analytic hydrostatic solution (a). Slower compression rates result in deeper folds (b).

the same total arc length but with different compression rates. We observe that the viscous resistance to shear in the fluid substrate at a high compression rate prevents the fold from plunging down. Eventually, the gravitational body force takes over and forces the membrane fold down. However, even after the point of self-contact is reached and the fold is fully formed, one can see in Figure 14(b) that lower compression rates result in deeper folds. This could have significant implications on the mechanics of lipid monolayers in lung surfactants and is the object of a future study.

### 5. CONCLUSION

In this paper, we presented an Eulerian formulation for the analysis of the large deformations of an immersed interface arising in a large number of soft matter mechanics problems. The model is versatile enough to accommodate various material behaviors from a hyperelastic membrane to a fluid interface. The introduction of a tangential velocity degree of freedom for the interface in combination with discontinuities in velocity and pressure fields enables us to simulate a wide variety of boundary conditions between the interface and the surrounding fluid while automatically satisfying the normal velocity continuity condition. These contributions were made numerically possible by coupling two very efficient methods: the X-FEM and the grid-based particle methods, which used in an Eulerian framework completely circumvents the problem of mesh distortion usually associated with Lagrangian finite elements and allows us to easily impose flux boundary conditions on both the fluid and the membrane. The combination of these methods has two key advantages: (1) the X-FEM makes it possible to account for field discontinuities within the finite element framework and (2) the grid-based particle method removes the need for time-consuming reinitialization, which is usually associated with the use of level-set in X-FEM. Furthermore, as opposed to the traditional level-set method, the grid-based particle method is very well suited to the computation of higher-order derivative on a curved surface and can accurately track the Lagrangian fields associated with an elastic interface. We showed, via a number of examples, that the approach gives very satisfactory results for a wide variety of problems involving the effect slip lengths, membrane elasticity, and characterized by very large shape changes. In this context, future research can describe various mechanisms that may include the phoretic transport of highly deformable cells [64], the diffusion of membrane associated molecules on curved manifolds and its coupling with mechanical deformation [65, 66], or the effect of slip length on vesicles tumbling under the effect of shear flow [42].

### APPENDIX: COMPONENTS OF THE TANGENT MATRIX

To obtain the discretized weak form of the governing equations, let us turn to the weak form (17) and decompose the integration over  $\Omega \cup \Gamma$  into the sum of the integration over the element domain  $\Omega_e$ :

$$\begin{aligned}
 \sum_{e=1}^{nel} & \left[ \int_{\Omega^e} (\mu \nabla \mathbf{w}_v : \nabla \mathbf{v} - p \nabla \mathbf{w}_v : \mathbf{I}) dV^e + \int_{\Gamma^e} \mathbf{w}_v \cdot \left( \frac{\mu}{l^+} \lambda^+ - \frac{\mu}{l^-} \lambda^- + \bar{\mathbf{f}} \right) ds^e \right. \\
 & + \int_{\Omega^e} w_p \nabla \cdot \mathbf{v} dV^e + \int_{\Gamma^e} w_p \lambda_p ds^e + \int_{\Gamma^e} \lambda_p ([p] + \bar{\mathbf{f}}^\perp) ds^e \\
 & + \int_{\Gamma^e} w_v^\parallel \left( \frac{\mu}{l^+} (\lambda^+ + [v]^{\parallel+}) + \frac{\mu}{l^-} (\lambda^- + [v]^{\parallel-}) + \bar{\mathbf{f}}^\parallel \right) ds^e \\
 & + \int_{\Gamma^e} w_{\lambda^+} \left( (\mu \nabla \mathbf{v} \cdot \bar{\mathbf{n}})^{\parallel+} - \frac{\mu}{l^+} [v]^{\parallel+} \right) ds^e \\
 & \left. + \int_{\Gamma^e} w_{\lambda^-} \left( (\mu \nabla \mathbf{v} \cdot \bar{\mathbf{n}})^{\parallel-} - \frac{\mu}{l^-} [v]^{\parallel-} \right) ds^e \right] = 0
 \end{aligned} \tag{A.1}$$

Next, we replace the different fields  $\mathbf{v}$ ,  $p$ ,  $\bar{v}^\parallel$ ,  $\lambda_p$ ,  $\lambda^+$ , and  $\lambda^-$  with their discretized form (18), (19) and (20). In order to minimize the length of the discretized governing equations, we first rewrite the interpolation equations as the product of a shape function matrix and the element nodal values vector as follows. Equation (19) can be rewritten as

$$\begin{aligned} \tilde{\mathbf{v}}^e(\mathbf{x}, t) &= \mathbf{N}^9(\mathbf{x}) \cdot \mathbf{v}^e(t) \\ \text{with } \mathbf{N}^9(\mathbf{x}) &= \begin{bmatrix} N_1^9 & 0 & N_9^9 & 0 \\ 0 & N_1^9 & \cdots & 0 & N_9^9 \\ (H - H_1)N_1^9 a_1^1 & 0 & (H - H_9)N_9^9 a_1^9 & 0 \\ 0 & (H - H_1)N_1^9 a_2^1 & \cdots & 0 & (H - H_9)N_9^9 a_2^9 \end{bmatrix} \\ \text{and } \mathbf{v}^e(t) &= \{v_1^1 \ v_2^1 \ v_1^2 \ v_2^2 \ \dots \ v_1^9 \ v_2^9, \hat{v}^1 \ \dots \ \hat{v}^9\}^T, \end{aligned}$$

while (18) takes the form:

$$\begin{aligned} \tilde{\mathbf{p}}^e(\mathbf{x}, t) &= \mathbf{N}^4(\mathbf{x}) \cdot \mathbf{p}^e(t) \\ \text{with } \mathbf{N}^4(\mathbf{x}) &= [N_1^4 \ \dots \ N_4^4 \ (H - H_1)N_1^4 \ \dots \ (H - H_4)N_4^4] \\ \text{and } \mathbf{p}^e(t) &= \{p^1 \ \dots \ p^4 \ \hat{p}^1 \ \dots \ \hat{p}^4\}^T, \end{aligned}$$

and (20) reads:

$$\begin{aligned} \tilde{\mathbf{v}}^{\parallel e}(\mathbf{x}, t) &= \bar{\mathbf{N}}(\mathbf{x}) \cdot \bar{\mathbf{v}}^{\parallel e}(t) \quad , \quad \tilde{\boldsymbol{\lambda}}_p^e(\mathbf{x}, t) = \bar{\mathbf{N}}(\mathbf{x}) \cdot \boldsymbol{\lambda}_p^e(t) \quad \text{and} \quad \tilde{\boldsymbol{\lambda}}^{\pm e}(\mathbf{x}, t) = \bar{\mathbf{N}}(\mathbf{x}) \cdot \boldsymbol{\lambda}^{\pm e}(t) \\ \text{with } \bar{\mathbf{N}} &= [\bar{N}_1 \ \bar{N}_2] \\ \text{and } \bar{\mathbf{v}}^{\parallel e}(t) &= \{\bar{v}^{\parallel 1} \ \bar{v}^{\parallel 2}\}^T \quad , \quad \boldsymbol{\lambda}_p^e(t) = \{\lambda_p^1 \ \lambda_p^2\}^T \quad , \quad \boldsymbol{\lambda}^{\pm e}(t) = \{\lambda^{\pm 1} \ \lambda^{\pm 2}\}^T \end{aligned}$$

We further write the rate of deformation of the fluid  $\mathbf{D}^e$  (in Voigt notation) and the velocity divergence  $\nabla \cdot \mathbf{v}^e$  in element  $e$  in the two-dimensional plane strain case as

$$\{\mathbf{D}^e\} = \begin{bmatrix} D_{11}^e \\ D_{22}^e \\ 2D_{12}^e \end{bmatrix} = \mathbf{B} \cdot \mathbf{v}^e \quad \text{and} \quad \nabla \cdot \mathbf{v}^e = \check{\mathbf{B}} \cdot \mathbf{v}^e$$

where the  $\mathbf{B}$  and  $\check{\mathbf{B}}$  matrices relate the nodal velocities to the deformation rate and velocity divergence:

$$\mathbf{B} = [\mathbf{B}_1 \ \mathbf{B}_2 \ \dots \ \mathbf{B}_{n_9+m_9}] \quad \text{with} \quad \mathbf{B}_I = \begin{bmatrix} \frac{\partial N_I^9(\mathbf{x})}{\partial x_1} & 0 \\ 0 & \frac{\partial N_I^9(\mathbf{x})}{\partial x_2} \\ \frac{\partial N_I^9(\mathbf{x})}{\partial x_2} & \frac{\partial N_I^9(\mathbf{x})}{\partial x_1} \end{bmatrix}$$

and

$$\check{\mathbf{B}} = [\check{\mathbf{B}}_1 \ \check{\mathbf{B}}_2 \ \dots \ \check{\mathbf{B}}_{n_9+m_9}] \quad \text{with} \quad \check{\mathbf{B}}_I = \begin{bmatrix} \frac{\partial N_I^9(\mathbf{x})}{\partial x_1} & \frac{\partial N_I^9(\mathbf{x})}{\partial x_2} \end{bmatrix}$$

with these definition, one can rewrite the weak form in its discretized form as follows:

$$\begin{aligned} &\sum_{e=1}^{nel} \{w_v^e\}^T \left[ \int_{\Omega_e} \mathbf{B}^T \cdot (\mu \mathbf{B} \cdot \mathbf{v}^e - \mathbf{N}^4 \cdot \mathbf{p}^e) dV^e \right. \\ &\quad \left. + \int_{\Gamma_e} (\mathbf{N}^9)^T \cdot \mathbf{a}_1 \left( \frac{\mu}{l^+} \bar{\mathbf{N}} \cdot \boldsymbol{\lambda}^{e+} - \frac{\mu}{l^-} \bar{\mathbf{N}} \cdot \boldsymbol{\lambda}^{e-} \right) ds^e + \int_{\Gamma_e} (\mathbf{N}^9)^T \cdot \bar{\mathbf{f}} ds^e \right] \\ &+ \sum_{e=1}^{nel} \{w_p^e\}^T \left[ \int_{\Omega_e} (\mathbf{N}^4)^T \cdot \check{\mathbf{B}} \cdot \mathbf{v}^e dV^e + \int_{\Gamma_e} (\mathbf{N}^4)^T \cdot \bar{\mathbf{N}} \cdot \boldsymbol{\lambda}_p^e ds^e \right] \\ &+ \sum_{e=1}^{nel} \{w_{\bar{v}}^e\}^T \left[ \int_{\Gamma_e} \bar{\mathbf{N}}^T \cdot \left( \frac{\mu}{l^+} \bar{\mathbf{N}} \cdot (\boldsymbol{\lambda}^{e+} + \bar{\mathbf{v}}^{\parallel e}) \right) - \frac{\mu}{l^+} \mathbf{N}_+^9 \cdot \mathbf{a}_1 \cdot \mathbf{v}^e \right] ds^e \end{aligned}$$

$$\begin{aligned} & \int_{\Gamma_e} \bar{\mathbf{N}}^T \cdot \left( \frac{\mu}{l^-} \bar{\mathbf{N}} \cdot (-\lambda^{e-} - \bar{\mathbf{v}}^{\parallel e}) + \frac{\mu}{l^-} \mathbf{N}_-^9 \cdot \mathbf{a}_1 \cdot \mathbf{v}^e \right) ds^e + \int_{\Gamma_e} \bar{\mathbf{N}}^T \cdot \bar{\mathbf{f}}^{\parallel} ds^e \Big] \\ & + \sum_{e=1}^{nel} \{w_{\lambda_p}^e\}^T \left[ \int_{\Gamma_e} \bar{\mathbf{N}}^T \cdot \mathbf{N}_{\square}^4 \cdot \mathbf{p}^e ds^e + \int_{\Gamma_e} \bar{\mathbf{N}}^T \cdot \bar{\mathbf{f}}^{\perp} ds^e \right] \\ & + \sum_{e=1}^{nel} \{w_{\lambda_+}^e\}^T \left[ \int_{\Gamma_e} \bar{\mathbf{N}}^T \cdot \left( \mu \mathbf{a}_1^T \cdot \mathbf{P}^{\perp} \cdot \mathbf{B}_+ \cdot \mathbf{v}^e - \frac{\mu}{l^+} (\bar{\mathbf{N}} \cdot \bar{\mathbf{v}}^{\parallel e} - \mathbf{N}_+^9 \cdot \mathbf{a}_1 \cdot \mathbf{v}^e) \right) ds^e \right] \\ & + \sum_{e=1}^{nel} \{w_{\lambda_-}^e\}^T \left[ \int_{\Gamma_e} \bar{\mathbf{N}}^T \cdot \left( \mu \mathbf{a}_1^T \cdot \mathbf{P}^{\perp} \cdot \mathbf{B}_- \cdot \mathbf{v}^e + \frac{\mu}{l^-} (\bar{\mathbf{N}} \cdot \bar{\mathbf{v}}^{\parallel e} - \mathbf{N}_-^9 \cdot \mathbf{a}_1 \cdot \mathbf{v}^e) \right) ds^e \right] = 0 \end{aligned}$$

where the subscript + and - on the  $\mathbf{B}$  and  $\mathbf{N}^9$  indicate whether the fluid velocity (and its derivative) is interpolated on the + or - side of  $\Gamma$  while  $\mathbf{N}_{\square}^4 = [0..0 \ N_1^4 \dots N_4^4]$  is used to interpolate the pressure jump across the interface. The aforementioned equation can then conveniently be written in the following format:

$$\begin{bmatrix} \mathbf{K}^t & \mathbf{I}_2^t \\ \mathbf{I}_1^t & \bar{\mathbf{K}}^t \end{bmatrix} \cdot \begin{bmatrix} \mathbf{d}(t) \\ \bar{\mathbf{d}}(t) \end{bmatrix} + \begin{bmatrix} \mathbf{F}^t \\ \bar{\mathbf{F}}^t \end{bmatrix} = 0$$

where the different submatrices are computed at each element as follows:

$$\mathbf{K}_e^t = \begin{bmatrix} \mathbf{K}_{vv}^e & \mathbf{K}_{vp}^e \\ \mathbf{K}_{pv}^e & 0 \end{bmatrix}, \quad \bar{\mathbf{K}}_e^t = \begin{bmatrix} \mathbf{K}_{\bar{v}\bar{v}}^e & 0 & \mathbf{K}_{\bar{v}\lambda_+}^e & \mathbf{K}_{\bar{v}\lambda_-}^e \\ 0 & 0 & 0 & 0 \\ \mathbf{K}_{\lambda_+\bar{v}}^e & 0 & 0 & 0 \\ \mathbf{K}_{\lambda_-\bar{v}}^e & 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{I}_{e1}^t = \begin{bmatrix} \mathbf{K}_{\bar{v}\bar{v}}^e & 0 \\ 0 & \mathbf{K}_{\lambda_p p}^e \\ \mathbf{K}_{\lambda_+v}^e & 0 \\ \mathbf{K}_{\lambda_-v}^e & 0 \end{bmatrix} \tag{A.2}$$

$$\mathbf{I}_{e2}^t = \begin{bmatrix} 0 & 0 & \mathbf{K}_{v\lambda_+}^e & \mathbf{K}_{v\lambda_-}^e \\ 0 & \mathbf{K}_{p\lambda_p}^e & 0 & 0 \end{bmatrix}, \quad \mathbf{F}_{fe}^t = \begin{bmatrix} \mathbf{F}_{\bar{v}}^e \\ \mathbf{F}_p^e \end{bmatrix} \text{ and } \bar{\mathbf{F}}_{fe}^t = \begin{bmatrix} \mathbf{F}_{\bar{v}\bar{v}}^e \\ \mathbf{F}_{\lambda_p}^e \\ 0 \\ 0 \end{bmatrix} \tag{A.3}$$

The various components of the element matrix  $\mathbf{K}^e$  are calculated as

$$\begin{aligned} \mathbf{K}_{vv}^e &= \int_{\Omega^e} \mu \mathbf{B}^T \cdot \mathbf{B} \, dV^e \\ \mathbf{K}_{vp}^e &= \int_{\Omega^e} -\mathbf{B}^T \cdot \mathbf{N}^4 \, dV^e \\ \mathbf{K}_{v\lambda_+}^e &= \int_{\Gamma_e} \frac{\mu}{l^+} (\mathbf{N}_-^9)^T \cdot \mathbf{a}_1 \cdot \bar{\mathbf{N}} \, ds^e \\ \mathbf{K}_{v\lambda_-}^e &= \int_{\Gamma_e} -\frac{\mu}{l^-} (\mathbf{N}_-^9)^T \cdot \mathbf{a}_1 \cdot \bar{\mathbf{N}} \, ds^e \\ \mathbf{K}_{pv}^e &= \int_{\Omega^e} (\mathbf{N}^4)^T \cdot \check{\mathbf{B}} \, dV^e \\ \mathbf{K}_{p\lambda_p}^e &= \int_{\Gamma_e} (\mathbf{N}_{\square}^4)^T \cdot \bar{\mathbf{N}} \, ds^e \\ \mathbf{K}_{\bar{v}\bar{v}}^e &= \int_{\Gamma_e} \bar{\mathbf{N}}^T \cdot \mathbf{a}_1 \cdot \left( \frac{\mu}{l^-} \cdot \mathbf{N}_-^9 - \frac{\mu}{l^+} \cdot \mathbf{N}_+^9 \right) ds^e \\ \mathbf{K}_{\bar{v}\bar{v}\bar{v}}^e &= \int_{\Gamma_e} \bar{\mathbf{N}}^T \cdot \left( \frac{\mu}{l^+} - \frac{\mu}{l^-} \right) \cdot \bar{\mathbf{N}} \, ds^e \\ \mathbf{K}_{\bar{v}\bar{v}\lambda_+}^e &= \int_{\Gamma_e} \frac{\mu}{l^+} \bar{\mathbf{N}}^T \cdot \bar{\mathbf{N}} \, ds^e \end{aligned}$$

$$\begin{aligned}
\mathbf{K}_{\bar{v}\|\lambda^-}^e &= \int_{\Gamma^e} -\frac{\mu}{l^-} \bar{\mathbf{N}}^T \cdot \bar{\mathbf{N}} ds^e \\
\mathbf{K}_{\lambda_p p}^e &= \int_{\Gamma^e} \bar{\mathbf{N}}^T \cdot \mathbf{N}_{[]}^4 ds^e \\
\mathbf{K}_{\lambda+v}^e &= \int_{\Gamma^e} \bar{\mathbf{N}}^T \cdot (\mu \mathbf{a}_1^T \cdot \mathbf{P}^\perp \cdot \mathbf{B}_+ + \mathbf{N}_+^9 \cdot \mathbf{a}_1) ds^e \\
\mathbf{K}_{\lambda+\bar{v}\|}^e &= \int_{\Gamma^e} -\frac{\mu}{l^+} \bar{\mathbf{N}}^T \cdot \bar{\mathbf{N}} ds^e \\
\mathbf{K}_{\lambda-v}^e &= \int_{\Gamma^e} \bar{\mathbf{N}}^T \cdot (\mu \mathbf{a}_1^T \cdot \mathbf{P}^\perp \cdot \mathbf{B}_- - \mathbf{N}_-^9 \cdot \mathbf{a}_1) ds^e \\
\mathbf{K}_{\lambda-\bar{v}\|}^e &= \int_{\Gamma^e} \frac{\mu}{l^-} \bar{\mathbf{N}}^T \cdot \bar{\mathbf{N}} ds^e
\end{aligned}$$

while the external forces associated with element  $e$  are written as

$$\begin{aligned}
\mathbf{F}_v^e &= \int_{\Omega^e} -(\mathbf{N}^9)^T \cdot \bar{\mathbf{f}} dV^e \\
\mathbf{F}_{\bar{v}\|}^e &= \int_{\Gamma^e} \bar{\mathbf{N}}^T \cdot \bar{\mathbf{f}}^{\|} ds^e \\
\mathbf{F}_{\lambda_p}^e &= \int_{\Gamma^e} \bar{\mathbf{N}}^T \cdot \bar{\mathbf{f}}^\perp ds^e
\end{aligned}$$

where  $\bar{\mathbf{f}}$  is computed using the updated strain and curvature. The computation of these quantities involves the assessment of integrals over elements that can be numerically evaluated using Gaussian quadrature with four integration points in regular elements. However, to carry out the integration over the two sides of  $\Gamma$  in split elements, a division into sub-triangles is required as described in [38].

#### REFERENCES

1. Block J, Stradal TEB, Hänisch J, Geffers R, Köstler SA, Urban E, Small JV, Rottner K, Faix J. Filopodia formation induced by active mDia2/Drf3. *Journal of Microscopy* 2008; **231**(3):506–517.
2. Ponti A, Machacek M, Gupton SL, Waterman-Storer CM, Danuser G. Two distinct actin networks drive the protrusion of migrating cells. *Science* 2004; **305**(5691):1782–6.
3. Farsad M, Vernerey FJ. An XFEM-based numerical strategy to model mechanical interactions between biological cells and a deformable substrate. *International Journal for Numerical Methods in Engineering* 2012; **92**(3):238–267.
4. Charras G, Paluch E. Blebs lead the way: how to migrate without lamellipodia. *Nature Reviews. Molecular Cell Biology* September 2008; **9**(9):730–6.
5. Tinevez JY, Schulze U, Salbreux G, Roensch J, Joanny JF, Paluch E. Role of cortical tension in bleb growth. *Proceedings of the National Academy of Sciences of the United States of America* 2009; **106**(44):18581–6.
6. Beck R, Bruegger B, Wieland FT. Membrane deformation and separation. *F1000 Biology Reports* 2010; **2**(May): 23–25.
7. McMahon HT, Gallop JL. Membrane curvature and mechanisms of dynamic cell membrane remodelling. *Nature* 2005; **438**(7068):590–6.
8. Walther TC, Brickner JH, Aguilar PS, Bernales S, Pantoja C, Walter P. Eisosomes mark static sites of endocytosis. *Nature* 2006; **439**(7079):998–1003.
9. Peskin C. Flow patterns around heart valves: a numerical method. *Journal of Computational Physics* 1972; **10**(2):252–271.
10. Peskin CS. Numerical analysis of blood flow in the heart.pdf. *Journal of Computational Physics* 1977; **25**(1): 220–252.
11. Peskin CS. The immersed boundary method. *Acta Numerica* 2002; **11**(1):1–39.
12. Wang X, Liu WK. Extended immersed boundary method using FEM and RKPM. *Computer Methods in Applied Mechanics and Engineering* March 2004; **193**(12–14):1305–1321.
13. Zhang L, Gerstenberger A, Wang Xiaodong. Immersed finite element method. *Computer Methods in Applied Mechanics and Engineering* 2002; **September 2003**:1–25.
14. Glowinski R, Pan TW, Periaux J. A fictitious domain method for external incompressible viscous flow modeled by Navier–Stokes equations. *Computer Methods in Applied Mechanics and Engineering* 1994; **112**(1–4):133–148.

15. Glowinski R, Pan T, Hesla TI, Joseph DD. A distributed Lagrange multiplier / fictitious domain method for particulate flows. *International Journal of Multiphase Flow* 1999; **25**:755–794.
16. van Loon R, Anderson PD, Baaijens FPT, van de Vosse FN. A three-dimensional fluid structure interaction method for heart valve modelling. *Comptes Rendus Mécanique* December 2005; **333**(12):856–866.
17. Baaijens FPT. A fictitious domain/mortar element method for fluid structure interaction. *International Journal for Numerical Methods in Fluid* 2001; **May 1999**:743–761.
18. Bazhlekov IB, Anderson PD, Meijer HEH. Nonsingular boundary integral method for deformable drops in viscous flows. *Physics of Fluids* 2004; **16**(4):1064.
19. Zinchenko AZ, Rother Ma, Davis RH. A novel boundary-integral algorithm for viscous interaction of deformable drops. *Physics of Fluids* 1997; **9**(6):1493.
20. Pozrikidis C. Flow-induced deformation of an elastic membrane adhering to a wall. *International Journal of Solids and Structures* 2009; **46**(17):3198–3208.
21. Wu JZ, Yang YT, Luo YB, Pozrikidis C. Fluid kinematics on a deformable surface. *Journal of Fluid Mechanics* 2005; **541**(-1):371.
22. Ma L, Klug WS. Viscous regularization and r-adaptive remeshing for finite element analysis of lipid membrane mechanics. *Journal of Computational Physics* 2008; **227**(11):5816–5835.
23. Cottet Georges-Henri, Maitre Emmanuel. A level set method for fluid–structure interactions with immersed surfaces. *Mathematical Models and Methods in Applied Sciences* 2006; **16**(03):415.
24. Sher SO. A level set formulation of Eulerian interface capturing methods for incompressible fluid flows 1996; **464**(124):449–464.
25. Duddu R, Bordas SPA, Moran B, Chopp D. A combined extended finite element and level set method for biofilm growth. *International Journal for Numerical Methods in Engineering* 2008; **74**(October 2007):848–870.
26. Udaykumar HS. *An Eulerian method for computation of multimaterial impact with ENO shock-capturing and sharp interfaces*, Vol. 186, 2003.
27. Okazawa S, Kashiyama K, Kaneko Y. Eulerian formulation using stabilized finite element method for large deformation solid dynamics. *International Journal for Numerical Methods in Engineering* 2007; **72**(13):1544–1559.
28. II S, Sugiyama K, Takeuchi S, Takagi S, Matsumoto Y. An implicit full Eulerian method for the fluid structure interaction problem. *International Journal for Numerical Methods in Fluids* 2011; **65**(1-3):150–165.
29. Cottet Gh, Maitre E, Milcent T. Eulerian formulation and level set models for incompressible fluid–structure interaction. *ESAIM Mathematical Modelling and Numerical Analysis* 2008; **42**(3):471–492.
30. Hoogstraten PAA, Slaats PMA, Baaijens FPT. A Eulerian approach to the finite element modelling of neo-Hookean rubber material. *Applied Scientific Research* 1991; **48**(1):193–210.
31. Liu C, Walkington NJ. An Eulerian description of fluids containing visco-elastic particles. *Archive for Rational Mechanics and Analysis* 2001; **159**(3):229–252.
32. Matthews MT, Hill JM. Micro/nano sliding plate problem with Navier boundary condition. *Zeitschrift für angewandte Mathematik und Physik* 2006; **57**(5):875–903.
33. Anh T, Papavassiliou DV, Lee LL, Striolo A. Liquid water can slip on a hydrophilic surface. *PNAS* 2011; **108**(39):16170–16175.
34. Helfrich W. Elastic properties of lipid bilayers: theory and possible experiments. *Zeitschrift für Naturforschung Teil C Biochemie Biophysik Biologie Virologie* 1973; **28**(11):693–703.
35. Arroyo M, DeSimone A. Relaxation dynamics of fluid membranes. *Physical Review E* 2009; **79**(3):031915-1–031915-17.
36. Wall WA, Gerstenberger A, K U. An XFEM based fixed-grid approach for 3D fluid–structure interaction. In *Fluid Structure Interaction ii*, vol. 73, Bungartz H-J, Mehl M, Schäfer M (eds), Lecture Notes in Computational Science and Engineering. Springer Berlin Heidelberg: Berlin, Heidelberg, 2010; 327–349.
37. Legay a, Chessa J, Belytschko T. An Eulerian Lagrangian method for fluid structure interaction based on level sets. *Computer Methods in Applied Mechanics and Engineering* March 2006; **195**(17-18):2070–2087.
38. Moës N, Dolbow J, Belytschko T. A finite element method for crack growth without remeshing. *International Journal for Numerical Methods in Engineering* 1999; **46**(1):131–150.
39. Moës N, Béchet E, Tourbier M. Imposing Dirichlet boundary conditions in the extended finite element method. *International Journal for Numerical Methods in Engineering* 2006; **67**(12):1641–1669.
40. Song JH, Areias PM, Belytschko T. A method for dynamic crack and shear band propagation with phantom nodes. *International Journal for Numerical Methods in Engineering* 2006; **67**(6):868–893.
41. Leung S, Zhao H. A grid based particle method for moving interface problems. *Journal of Computational Physics* 2009; **228**(8):2993–3024.
42. Sui Y, Chew Y, Roy P, Chen X, Low H. Transient deformation of elastic capsules in shear flow: effect of membrane bending stiffness. *Physical Review E* 2007; **75**(6):066301-1–066301-10.
43. Vernerey FJ. A theoretical treatment on the mechanics of interfaces in deformable porous media. *International Journal of Solids and Structures* November 2011; **48**(22-23):3129–3141.
44. Bathe KJ. *Finite Element Procedures*. Prentice Hall: Englewood Clis, N.J, 1996.
45. Belytschko T, Parimi C, Moës N, Sukumar N, Usui S. Structured extended finite element methods for solids defined by implicit surfaces. *International Journal for Numerical Methods in Engineering* 2003; **56**(4):609–635.
46. Béchet1 É, Moës N, Wohlmuth B. A stable Lagrange multiplier space for stiff interface conditions within the extended finite element method. *IJNME* 2009; **78**(8):931–954.

47. Fries T, Belytschko T. The extended/generalized finite element method: an overview of the method and its applications 2010; **April**:253–304.
48. Babuška I, Banerjee U. Stable generalized finite element method (SGFEM). *Computer Methods in Applied Mechanics and Engineering* 2012; **201-204**:91–111.
49. Béchet E, Minnebo H, Moës N, Burgardt B. Improved implementation and robustness study of the X-FEM for stress analysis around cracks. *International Journal for Numerical Methods in Engineering* 2005; **64**(8):1033–1056.
50. Menk A, Bordas SPA. A robust preconditioning technique for the extended finite element method 2011; **October 2010**:1609–1632.
51. Sauerland H, Fries T. The stable XFEM for two-phase flows. *Computers and Fluids* 2012.
52. Enright D, Losasso F, Fedkiw R. A Fast and accurate semi-Lagrangian particle level set method 2004:1–24.
53. Jason H, Kumar T. *Advances in Computational Dynamics of Particles, Materials and Structures*. John Wiley & Sons, Ltd: New York, N.Y., 2012.
54. Pop JJ. Acceleration waves in isotropic elastic membranes. *Archive for Rational Mechanics and Analysis* 1981; **77**(1):47–93.
55. Landau LD, Lifshitz EM. *Fluid Mechanics (2nd Ed)*. Pergamon Press: Oxford, England, 1987.
56. Hughes TJR, Masud A, Wan J. A stabilized mixed discontinuous Galerkin method for Darcy flow. *Computer Methods in Applied Mechanics and Engineering* 2006; **195**(25-28):3347–3381.
57. Cheng KW, Fries T. Higher-order XFEM for curved strong and weak discontinuities 2010; **November 2009**: 564–590.
58. Leung S, Lowengrub J, Zhao H. A grid based particle method for solving partial differential equations on evolving surfaces and modeling high order geometrical motion. *Journal of Computational Physics* 2011; **230**(7):2540–2561.
59. Schürch S. Surface tension at low lung volumes: dependence on time and alveolar size. *Respiration Physiology* 1982; **48**(3):339–355.
60. Diamant H, Witten T, Ege C, Gopal a, Lee KY. Topography and instability of monolayers near domain boundaries. *Physical Review. E, Statistical, Nonlinear, and Soft Matter Physics* 2001; **63**(6 – Pt 1):061602-1–061602-12.
61. Diamant H, Witten TA. Compression induced folding of a sheet: an integrable system. *Physical Review Letters* 2011; **107**(16):68–81.
62. Hu JG, Granek R. Buckling of amphiphilic monolayers induced by head-tail asymmetry. *Journal de Physique II* 1996; **6**(7):999–1022.
63. Lu W, Knobler CM, Bruinsma RF, Twardos M, Dennin M. Folding Langmuir monolayers. *Physical Review Letters* 2002; **89**(14):146107-1–146107-4.
64. Anderson JL. Colloid transport by interfacial forces. *Annual Review of Fluid Mechanics* 1989; **21**(1):61–99.
65. Sun M, Northup N, Marga F, Huber Tams, Byfield FJ, Levitan I, Forgacs Gabor. The effect of cellular cholesterol on membrane–cytoskeleton adhesion. *Journal of Cell Science* 2007; **120**(Pt 13):2223–2231.
66. Foucard L, Vernerey FJ. A thermodynamical model for stress-fiber organization in contractile cells. *Applied Physics Letters* 2012; **100**(1):13702–137024.