



HAL
open science

An unstructured solver for simulations of deformable particles in flows at arbitrary Reynolds numbers

Simon Mendez, Etienne Gibaud, Franck Nicoud

► **To cite this version:**

Simon Mendez, Etienne Gibaud, Franck Nicoud. An unstructured solver for simulations of deformable particles in flows at arbitrary Reynolds numbers. *Journal of Computational Physics*, 2014, 256 (1), pp.465-483. 10.1016/j.jcp.2013.08.061 . hal-00871557

HAL Id: hal-00871557

<https://hal.science/hal-00871557>

Submitted on 9 Oct 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

An unstructured solver for simulations of deformable particles in flows at arbitrary Reynolds numbers

S. Mendez^{*}, E. Gibaud, F. Nicoud,

I3M, CNRS and University Montpellier II, 34095 Montpellier, France.

Abstract

As a step in the development of a numerical procedure able to perform parallel computations of the dynamics of capsules/cells in non-physiological configurations, a numerical method is developed and its validation is described. The fluid-structure interaction problem is solved using an immersed boundary method, adapted to an unstructured finite-volume flow solver thanks to the reproducing kernel particle method. A specific treatment to ensure volume conservation of the fluid enclosed in the immersed structure is also detailed. The present paper focuses on quantitative validation of the method in 2-D, against existing reference 2-D results. Excellent agreement is obtained for configurations of capsules and vesicles evolving with or without mean flow. Applications of the method to non-zero Reynolds number cases, including non-trivial geometry, is shown. This unstructured immersed boundary method proves robust to tackle the dynamics of deformable particles in complex flows.

Key words:

Capsule/cell dynamics, Unstructured solver, immersed boundary method, validation

1 MOTIVATION AND OBJECTIVES

Over the last decades, numerical simulation of capsules, vesicles and cells under flow has developed tremendously. All these systems, which will be referred to as deformable particles, are constituted by a liquid droplet enclosed by a very thin structure (its thickness is much smaller than the size of the object). This structure can be a polymer structure for capsules, a phospholipid bilayer for vesicles or a more complex biological membrane in the case of red blood cells [1]. Computing the dynamics of these deformable objects in flows is usually done

^{*} Corresponding author.

Address: University Montpellier II, I3M, UMR CNRS 5149, CC 051, 34095 Montpellier, France.

Email address: smendez@um2.fr (S. Mendez).

by coupling fluid equations with the relevant structure equations and by assuming an infinitely thin massless structure/membrane [1–5]. The flexible structure will be referred to as membrane in the following.

The most mature methods for computing flows of these deformable particles are based on boundary integral methods (BIM). Capsules, vesicles and red blood cells are small particles (of typical size of a few microns) and are generally carried by slow flows. The Reynolds number associated with many practical situations is so small that the flow is well described by the Stokes equations. Stokes equations are linear and the boundary integral method [6] can be used to predict the fluid velocity everywhere, knowing the state of the membrane. This reduces the fluid-structure interaction (FSI) to a problem where the only unknowns are the membrane coordinates over time. Boundary integral method is thus a very popular technique to compute flows of capsules [4, 5, 7–11], vesicles [12–16] and red blood cells [17–19], because of its precision and its relatively moderate computational cost.

The drawback of boundary integral methods is their limitation to Stokes flows. More recently, several methods have been adapted to the computation of deformable particles without this limitation to very small Reynolds number. Specific particle-based methods have successfully computed the dynamics of vesicles and red blood cells in various flows [20–23]. Such methods predict the fluid flow by solving equations of motion for particles representing clusters of molecules or atoms. They are generally coupled to another particle-based formalism to represent the membrane. Recently, lattice-Boltzmann methods have also been adapted to compute flows of red blood cells [24, 25]. Regarding Eulerian formalisms for the fluid flow, methods with the fluid grid following the deformation of the interface, based on the Arbitrary Lagrangian-Eulerian formalism are developed [26, 27]. However, they are not the most popular, as they involve frequent remeshing. Authors generally prefer one-fluid formalisms, where the fluid equations are solved everywhere, over a fixed Eulerian grid. The membrane location is computed by advecting either a function as the level-set function or a second Lagrangian grid (typically, a triangulation of the membrane) following the membrane displacement. One can cite the advected-field approach [28, 29], level-set methods [30–33] and immersed boundary or immersed interface methods [34–41]. In particular, the immersed boundary method has proven to be mature, providing 3-D results for the dynamics of capsules [39–44]. Recently, the immersed boundary method has also been used to study particle interactions [40] and relatively dense suspensions [43]. Note that the immersed boundary method can be used in the lattice-Boltzmann context to account for the fluid-structure coupling [24].

Immersed boundary methods (IBM) are used, in general, in conjunction with a regular Cartesian grid. However, for some applications, it might be useful to rely on unstructured methods for solving the fluid equations to account for complex geometries accurately. This is notably the case when dealing with medical artificial devices which usually involve complex manufactured geometries, possibly evolving over time (as in ventricular assist devices). Flow computations in such systems are challenging not only for their geometrical complexity but also for the high flow speed encountered: flow speeds and Reynolds number are often much larger in artificial devices than in the circulatory system [45]. Optimization of such systems (for instance to prevent hemolysis or thrombosis) thus necessitates methods able to account accurately for 1) presence

of biological cells, 2) high Reynolds number flows and 3) complex geometries. In addition, one could also take advantage of an unstructured method to refine the fluid grid locally, for instance in near-wall regions to better describe the cell-wall interactions.

The first objective of the paper is to address the question of how to adapt an existing unstructured finite-volume flow solver to the simulation of the dynamics of deformable objects under flow (note that a similar approach has been undertaken by Liu and co-workers, to adapt the IBM to the finite element context [36, 46–48]). The method is detailed in section 2. As a first step towards full 3-D calculations, the 2-D implementation is detailed in the present study. Two-dimensional implementation has the advantage of focusing the attention on the coupling between the fluid flow and the membrane dynamics. Specific 3-D difficulties are more related with the calculation of membrane force over a discretized 3-D surface than with the fluid-structure coupling. As already stated, using such a method in the context of deformable particles is not standard. As a consequence, special attention is paid to the validation of the solver, which is the object of section 3. Careful validation of our 2-D solver is the second objective of the paper: validation is indeed not that simple in 2-D. This comes from the fact that most of the reference results (either from experiments or BIM) are in 3-D. The moderate cost of BIM makes 3-D simulations have a relatively low computational cost. This explains why validations of solvers predicting the dynamics of deformable particles is often only qualitative. Nevertheless, BIM simulations in 2-D are rare, but some results are available in the open literature [12, 14, 49, 50]. They will be of primary help for validation of the present solver. Finally, the last section shows some applications beyond the test cases usually reported in the literature: section 4 shows two cases at non-zero Reynolds number. First, the influence of the Reynolds number is shown for a capsule suspended in a simple shear flow. Then, the evolution of a capsule deposited in a contraction flow at main flow Reynolds number 220 is computed.

2 NUMERICAL METHOD

2.1 Modeling framework

The present numerical method is dedicated to the numerical simulation of the dynamics of deformable particles, composed by an internal fluid enclosed by a flexible structure. The fluid inside and outside the cell is supposed to be incompressible and Newtonian. The fluid flow is thus governed by the incompressible versions of the continuity and the Navier-Stokes equations:

$$\nabla \cdot \mathbf{u} = 0, \tag{1}$$

$$\rho \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) = -\nabla p + \nabla \cdot [\mu (\nabla \mathbf{u} + (\nabla \mathbf{u})^T)], \tag{2}$$

where \mathbf{u} is the fluid velocity, p the pressure and t the time. The dynamic viscosity μ can be different for the internal fluid and the external fluid. Density variations are neglected, so that

ρ takes the same value inside and outside the membrane.

The flexible structure, or membrane, is an infinitely thin massless structure, completely closed, and transported by the fluid. In dimension n , the membrane is a structure of dimension $n - 1$ (a surface in 3-D), denoted by S (see Fig. 1a). Given a membrane position, it is possible to define a structural membrane energy \mathcal{E} , composed by an elastic component \mathcal{E}_e and a bending component \mathcal{E}_b : $\mathcal{E} = \mathcal{E}_e + \mathcal{E}_b$. The elastic component can be taken as:

$$\mathcal{E}_e = \int_S T dS, \quad (3)$$

where T is the surface tension in the membrane. It is related (see further) to E_e , the membrane elastic modulus (in Pa.m).

The bending part is:

$$\mathcal{E}_b = \frac{E_b}{2} \int_S \kappa^2 dS, \quad (4)$$

where E_b is the membrane bending modulus (in Pa.m³), κ the curvature (inverse of the radius of curvature; with \mathbf{n} the outward pointing normal vector to the membrane, $\kappa = \nabla \cdot \mathbf{n}$). This energy corresponds to the bending energy proposed by Helfrich (1973) [51], in its version for 2-D closed membranes. Spontaneous curvature is supposed to be zero. The formula with spontaneous curvature can be found in numerous references [3, 52].

By calculating the functional derivative of the energy with respect to an infinitesimal deformation of the membrane [29], it is possible to derive the force per unit surface that the membrane exerts on the fluid. The elastic and the curvature components, \mathbf{F}^{elas} and \mathbf{F}^{bend} , are separated (unit outward normal vector \mathbf{n} and tangential vector $\boldsymbol{\tau}$ are displayed in Fig. 1a):

$$\mathbf{F}^{elas} = -\kappa T \mathbf{n} + \frac{\partial T}{\partial s} \boldsymbol{\tau}, \quad (5)$$

with s the arc length along the interface, measured in the direction of $\boldsymbol{\tau}$ [49].

$$\mathbf{F}^{bend} = E_b \left[\frac{1}{2} \kappa^3 + \Delta_{LB} \kappa \right] \mathbf{n}, \quad (6)$$

where Δ_{LB} is the Laplace-Beltrami operator, *ie*: $\Delta_{LB} = (\mathbf{I}_s \nabla) \cdot (\mathbf{I}_s \nabla)$ and $\mathbf{I}_s = \mathbf{I} - \mathbf{n}\mathbf{n}$ [3].

To couple the movements of the fluid and of the membrane, one must have continuity of the velocity at the interface, while fluid stress undergoes a discontinuity related to the membrane forces. With $[q] = q^{ext} - q^{int}$ the jump of a quantity q between its external value q^{ext} and internal value q^{int} , these conditions read: $[\mathbf{u}] = 0$ and $[\boldsymbol{\sigma}] \cdot \mathbf{n} = -(\mathbf{F}^{elas} + \mathbf{F}^{bend})$ [49], where $\boldsymbol{\sigma}$ is the stress tensor at the interface.

2.2 Fluid-structure coupling with the immersed boundary method

The present framework relies on the immersed boundary method (IBM), developed by Peskin and co-workers [34, 53, 54] and the front-tracking (FT) method [55]. First, the membrane movement is described in a Lagrangian manner. Membrane is simply transported by the fluid, and at the same velocity. Adherence of the fluid over the membrane makes fluid velocity continuous at the membrane location, and the membrane motion is described by:

$$\frac{d\mathbf{X}(t)}{dt} = \mathbf{U}(\mathbf{X}(t), t) = \int_{\Omega} \mathbf{u}(\mathbf{x}, t) \delta(\mathbf{x} - \mathbf{X}(t)) d\mathbf{x}. \quad (7)$$

\mathbf{X} denotes the membrane coordinates and \mathbf{U} the membrane velocity. Ω is the fluid domain: recall that since the IBM is a one-fluid formalism, Ω is the union of the internal and external fluid domains. The velocity of the membrane is calculated from the fluid velocity field, using the Dirac function δ , which ensures the continuity of velocities at the interface.

The effect of the particle deformation on the fluid motion is twofold: 1) fluid properties are variable in space (e.g. viscosity may differ inside and outside the membrane) and 2) as a reaction against its deformation, the membrane exerts a force \mathbf{f} on the fluid, so that the momentum conservation equation reads:

$$\rho \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) = -\nabla p + \nabla \cdot [\mu(\nabla \mathbf{u} + (\nabla \mathbf{u})^T)] + \mathbf{f}. \quad (8)$$

The force per unit volume \mathbf{f} is obtained from the membrane forces $\mathbf{F} = \mathbf{F}^{elas} + \mathbf{F}^{bend}$, through

$$\mathbf{f}(\mathbf{x}, t) = \int_S \mathbf{F}(\mathbf{X}(t), t) \delta(\mathbf{x} - \mathbf{X}(t)) dS. \quad (9)$$

The addition of this source term to the fluid momentum equation is a way of ensuring the correct stress discontinuity across the membrane.

2.3 Discretization of the coupled problem

The discretized problem is represented in Fig. 1(b). The membrane is discretized using a set of M Lagrangian markers. The discrete unknowns are thus the markers locations $\mathbf{X}_m = (X_m, Y_m)^t$ ($m = 1, \dots, M$) and Eq. (7) is advanced for each marker. The markers transport equation is advanced explicitly, using an explicit Euler scheme.

Fluid equations are discretized over unstructured grids. Discretization of the fluid equations leads to a transport equation for the membrane markers involving discrete Dirac functions to interpolate the fluid velocity at the markers location $\mathbf{x}_n = (x_n, y_n)^t$ ($n = 1, \dots, N$):

$$\mathbf{U}_m(t) = \sum_{n=1}^N \mathbf{u}_n(\mathbf{x}_n, t) \Delta_m(\mathbf{x}_n - \mathbf{X}_m(t)) dV_n, \quad (10)$$

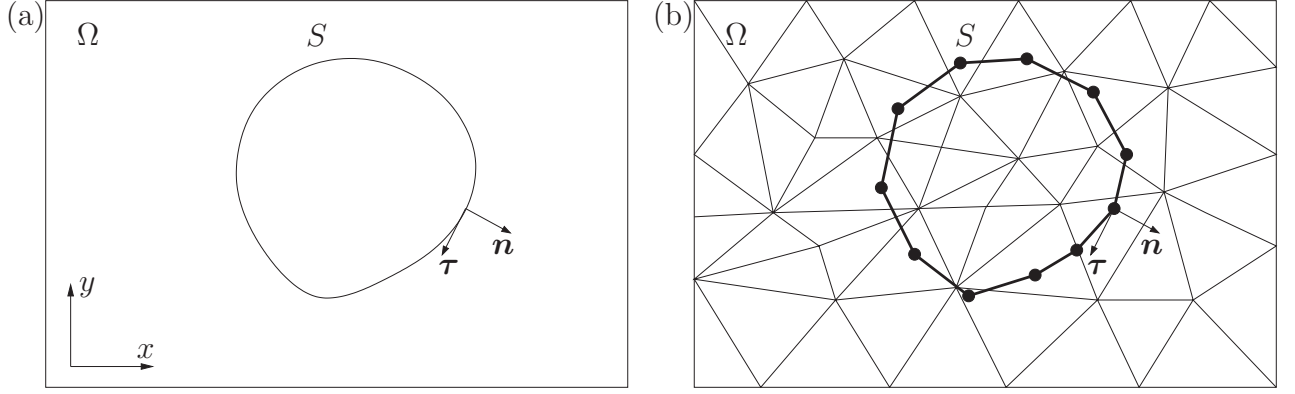


Fig. 1. Schematic of the physical problem (a) and of the IBM principle (b). The grid of the fluid domain Ω is represented as a triangular mesh. The fluid grid is fixed. Markers of the membrane S are shown by dots, linked by bold lines. This surface grid follows the membrane surface.

The grid contains N vertices; dV_n is the volume of the element associated with grid vertex n . The definition of the discrete Dirac function Δ_m is the object of next section.

The force per unit volume applied on the fluid by the membrane is calculated from the discrete forces calculated at the markers. This is called force regularization.

$$\mathbf{f}_n(t) = \sum_{m=1}^M \mathbf{F}_m(\mathbf{X}_m(t), t) dS_m \Delta_m(\mathbf{x} - \mathbf{X}_m(t)). \quad (11)$$

dS_m is the surface element (line in 2-D) associated with marker m and Δ_m denotes a discrete Dirac function, detailed later.

\mathbf{F}_m is the discrete force per unit surface applied by the membrane on the fluid around marker m . Calculus of the 2-D discrete elastic and curvature forces (Eqs. 5 and 6) is now described.

The elastic forces are calculated following the original IBM procedure by Peskin (1977) [53]. To approximate Eq. (5), the membrane is seen as a collection of links (the membrane edges) connecting the markers. The surface tension T is the elastic tension due to deformation of the elastic links: $T = E_e \epsilon$, where ϵ is the membrane strain in the direction of the membrane tangent [49]. More sophisticated elastic energy formulations (and subsequent tension formulae) can also be used, for example by adapting hyperelastic strain energy functions to 2-D [56]. Figure 2(a,b) shows the principle of the method. The reference (unstressed) configuration and the current configuration are displayed in Fig. 2(a) and Fig. 2(b), respectively. Strain is calculated for each edge m , by comparing the edge length l_m with a reference edge length l_m^{ref} representing the zero-stress configuration. The tension in each edge T_m is equal to

$$T_m = E_e \epsilon_m = E_e \frac{l_m - l_m^{ref}}{l_m^{ref}}. \quad (12)$$

One obtains the elastic membrane force ($\mathbf{F}_m^{elas} dS_m$) at marker m by summing the force contributions coming from the two edges (labelled $m-1$ and m) connecting marker m to its

neighbors:

$$\mathbf{F}_m^{elas} dS_m = T_m \frac{\mathbf{X}_{m+1} - \mathbf{X}_m}{\|\mathbf{X}_{m+1} - \mathbf{X}_m\|} + T_{m-1} \frac{\mathbf{X}_{m-1} - \mathbf{X}_m}{\|\mathbf{X}_{m-1} - \mathbf{X}_m\|}. \quad (13)$$

More sophisticated approximations can be employed, for example using cubic spline interpolation as in [49]. Note that such a modeling tolerates negative edge tension.

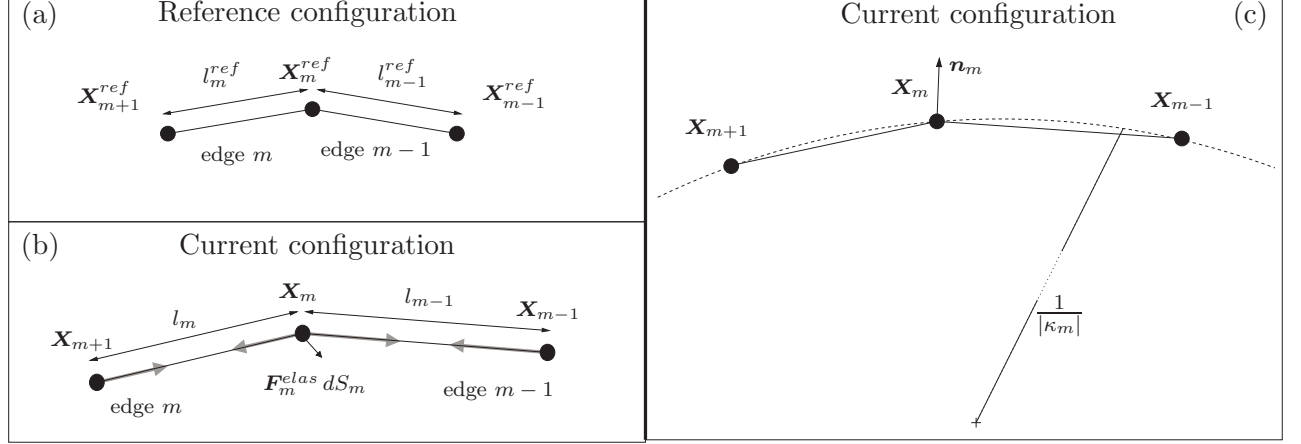


Fig. 2. Schematics for elastic force (left) and curvature (right) calculation. Elastic forces: differences in the length of the membrane edges between the reference unstressed configuration (a) and the current configuration (b) enables the calculation of the tension in each edge (grey arrows in sub-figure b). Contributions from the edges are summed at the vertices to calculate the elastic forces at the markers, $\mathbf{F}_m^{elas} dS_m$. Curvature: the circumscribed circle passing through marker m and its two neighbors is calculated. The inverse of the radius gives the curvature κ_m .

Numerically, curvature is calculated by finding the circumscribed circle of the triangle formed by each marker and its two neighbors (Fig. 2c). The inverse of the radius of the circumscribed circle is the curvature, in absolute value. With the definition $\kappa = \nabla \cdot \mathbf{n}$, the curvature κ_m is positive in Fig. 2(c). Using the circumscribed circle is valid when the local radius of curvature of the membrane is large enough compared to the edges length, which is always verified here. Numerical forces are simple to calculate in 2D. The Laplace-Beltrami operator is calculated by simple second-order finite differences accounting for non-uniform membrane mesh.

To perform computations of flows of deformable particles, the present IBM is implemented in an existing flow solver, YALES2 (<http://www.coria-cfd.fr/index.php/YALES2>). YALES2 is a finite-volume research code that solves the Navier-Stokes equations on unstructured meshes. It is dedicated to massively parallel computations of turbulent flows [57–59]. The incompressible Navier-Stokes equations are solved using a projection method [60]. The fluid velocity is first advanced using a 4th-order central scheme in space, and a 4th-order Runge-Kutta like scheme in time (see Moureau *et al.* [59] for more details). The new solver, gathering YALES2 and the IBM, is named YALES2BIO. In YALES2BIO, the forces exerted by the membrane on the fluid are included in the prediction step. Divergence-free velocity is obtained at the end of the time-step by solving a Poisson equation for pressure, to correct the predicted velocity. A Deflated Preconditioned Conjugate Gradient algorithm is used to solve this Poisson equation [57]. The correction step is not modified by the IBM. The fluid velocity at the end of each time step is interpolated to the markers, to convect the membrane.

2.4 Interpolation and regularization over unstructured meshes

In the original IBM formulation, as in a majority of studies using IBM, finite differences are used to solve the fluid equations, using a regular Cartesian grid of constant grid size h . In this case, the discrete Dirac function Δ_m involved in the velocity interpolation (Eq. 10) and force regularization (Eq. 11) can be easily defined. Δ_m is taken as a product of one-dimensional delta functions, $\Delta_m(\mathbf{x}_n - \mathbf{X}_m) = D(x_n - X_m) D(y_n - Y_m)$. The cosine representation is often used:

$$D(r) = \begin{cases} \frac{1}{4h} \left[1 + \cos\left(\frac{\pi r}{2h}\right) \right] & \text{if } |r| < 2h \\ 0 & \text{if } |r| \geq 2h \end{cases}. \quad (14)$$

The volume of the grid cells (involved in Eq. 10) is also defined in terms of grid size: $dV_n = h^3$. Note that more sophisticated delta functions were also proposed: see for example the discussions by Roma *et al.* [54] and Peskin [34] about the discrete Dirac functions and their properties. Over regular Cartesian grids, the discrete Dirac functions can be defined independently of the position of the markers.

Over unstructured grids, Cartesian versions of Δ_m cannot be used. This issue was dealt with in the context of finite elements by Liu and co-workers [36, 46–48] and their ideas were adapted later to the finite difference/finite volume context by Pinelli *et al.* [61]. Adaptation of the immersed boundary formalism to unstructured meshes rely on the Reproducing Kernel Particle Method (RKPM) [62].

The principle is briefly explained in 1D before detailing the implementation in the 2-D case. Based on an initial window function ϕ as the cosine one (Eq. 14; in that case $\phi(r) = D(r)$), one can introduce the following modified window function $\bar{\phi}$:

$$\bar{\phi}(X, x - X) = \sum_{k=0}^K \beta_k(X) (x - X)^k \phi(x - X). \quad (15)$$

X would be the marker coordinate and x the coordinate where the weight is to be calculated. β_k are the $K + 1$ coefficients of the polynomial correction of the original window function (the value of K is determined hereafter). They are calculated by imposing conditions on the first moments of $\bar{\phi}$. Defining the p^{th} moment of function $\bar{\phi}$ as :

$$\bar{m}_p(X) = \int_{\Omega} (x - X)^p \bar{\phi}(X, x - X) dx, \quad (16)$$

one easily obtains that

$$\bar{m}_p(X) = \sum_{k=0}^K \beta_k(X) m_{(k+p)}(X), \quad (17)$$

with $m_{(k+p)}(X)$ the $(k + p)^{\text{th}}$ moment of ϕ at the location of the marker, X . For a point force applied at coordinate X , moments calculated at X are known: the first moment is 1 and the

following are all 0. These conditions, which are naturally true for the force to regularize, are imposed on the regularized force. Verification of the moments conditions ensures the quality of the regularization process. In 1D, imposing that $\overline{m}_0(X) = 1$ and $\overline{m}_p(X) = 0$ for $1 \leq p \leq P$, one can determine the corresponding β_k for $0 \leq k \leq K = P$, by inverting a $(P + 1) \times (P + 1)$ symmetric matrix [36, 61]. When discrete Dirac functions are used for regularization of a marker force, having $\overline{m}_0(X) = 1$ and $\overline{m}_1(X) = 0$ guarantees that the regularized force has the same intensity and the same application point as the marker force.

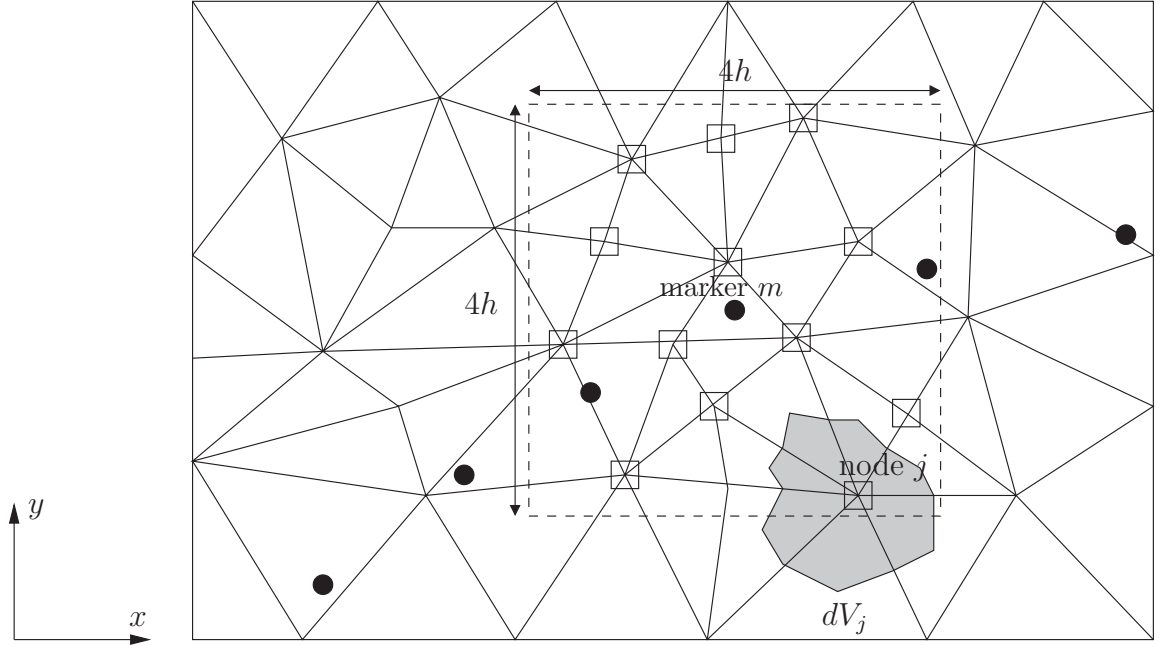


Fig. 3. Schematics of the procedure to compute the window function for regularization/interpolation over unstructured fluid grids. Membrane markers are shown by black dots, as in Fig. 1 (membrane edges are not represented to lighten the figure). Square symbols are used to show the J fluid grid nodes involved in the regularization of the force known at marker m : they are all enclosed in a square of edge length $4h$ around marker m . For node j , the associated cell volume dV_j is also shown.

Extension to two ¹ and three dimensions is shown in Pinelli *et al.* [61]. In practice, the following 2-D algorithm is used to calculate the weights of the window function associated with each marker. Figure 3 clarifies the regularization algorithm. For one marker m , the algorithm is:

- (1) First, the local grid size h is defined. A unique value, typical of the fluid grid size at the particle location, is used. Different values for each marker could be used, but have not been tested.
- (2) For all the J fluid nodes located in the square of edge length $4 \times h$ centered on marker m , the intermediate window function weights $w(\mathbf{X}_m, \mathbf{x}_j) = D(X_m - x_j) D(Y_m - y_j)$ are calculated ($j = 1, \dots, J$).

¹ Note however that typos remain in the 2-D matrix (their Eq. (28) [61]), which can be identified in the last column, by lack of symmetry.

(3) Moments of the intermediate window function are calculated as:

$$m_{p,q}(\mathbf{X}_m) = \sum_{j=1}^J (x_j - X_m)^p (y_j - Y_m)^q w(\mathbf{X}_m, \mathbf{x}_j) dV_j, \quad (18)$$

where dV_j is the area of the element associated with vertex j .

(4) In 2-D, moments of the intermediate window function are calculated as:

$$\overline{m}_{p,q}(\mathbf{X}_m) = \sum_{j=1}^J (x_j - X_m)^p (y_j - Y_m)^q \overline{w}(\mathbf{X}_m, \mathbf{x}_j) dV_j. \quad (19)$$

For the modified window function to be a good regularization of a point force, the values of moments of order 0, 1 and 2 are imposed: $\overline{m}_{0,0} = 1$ and $\overline{m}_{p,q} = 0$, if $1 \leq p + q \leq 2$. This necessitates, for each marker, the use of 6 coefficients $(\beta_k)_m$ ($0 \leq k \leq 5$). In order to calculate them, the 2-D moments matrix is built (the dependence of moments on \mathbf{X}_m is omitted):

$$\mathcal{M}_{2D} = \begin{pmatrix} m_{0,0} & m_{1,0} & m_{0,1} & m_{1,1} & m_{2,0} & m_{0,2} \\ m_{1,0} & m_{2,0} & m_{1,1} & m_{2,1} & m_{3,0} & m_{1,2} \\ m_{0,1} & m_{1,1} & m_{0,2} & m_{1,2} & m_{2,1} & m_{0,3} \\ m_{1,1} & m_{2,1} & m_{1,2} & m_{2,2} & m_{3,1} & m_{1,3} \\ m_{2,0} & m_{3,0} & m_{2,1} & m_{3,1} & m_{4,0} & m_{2,2} \\ m_{0,2} & m_{1,2} & m_{0,3} & m_{1,3} & m_{2,2} & m_{0,4} \end{pmatrix} \quad (20)$$

and inverted for each marker. The coefficients $(\beta_k)_m$ are then calculated using

$$\begin{pmatrix} (\beta_0)_m \\ (\beta_1)_m \\ (\beta_2)_m \\ (\beta_3)_m \\ (\beta_4)_m \\ (\beta_5)_m \end{pmatrix} = \mathcal{M}_{2D}^{-1} \begin{pmatrix} \overline{m}_{0,0} \\ \overline{m}_{1,0} \\ \overline{m}_{0,1} \\ \overline{m}_{1,1} \\ \overline{m}_{2,0} \\ \overline{m}_{0,2} \end{pmatrix} = \mathcal{M}_{2D}^{-1} \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}. \quad (21)$$

(5) The weights of the modified window function can thus be calculated using

$$\begin{aligned} \overline{w}(\mathbf{X}_m, \mathbf{x}_j) = w(\mathbf{X}_m, \mathbf{x}_j) [& (\beta_0)_m + (\beta_1)_m(x_j - X_m) \\ & + (\beta_2)_m(y_j - Y_m) + (\beta_3)_m(x_j - X_m)(y_j - Y_m) \\ & + (\beta_4)_m(x_j - X_m)^2 + (\beta_5)_m(y_j - Y_m)^2]. \end{aligned} \quad (22)$$

These weights are used to regularize the membrane forces on the fluid grids and for interpolation of the fluid velocity to the markers, using :

$$\mathbf{f}_n(t) = \sum_{m=1}^M \mathbf{F}_m(\mathbf{X}_m(t), t) dS_m \bar{w}(\mathbf{X}_m, \mathbf{x}_n), \quad (23)$$

$$\mathbf{U}_m(t) = \sum_{n=1}^N \mathbf{u}_n(\mathbf{x}_n, t) \bar{w}(\mathbf{X}_m, \mathbf{x}_n) dV_n. \quad (24)$$

Note that the window function \bar{w} depends both on the fluid grid *and* on the markers. To avoid recomputing the weights without storing them (their size being $N \times M$), $(\beta_k)_m$ (data of size $6 \times M$) are stored after the regularization step.

Note that this algorithm has been modified by Pinelli *et al.* to avoid ill-conditioned \mathcal{M}_{2D} matrices [61]. Such an issue has not been encountered in the present study, presumably because of the larger stencil used here ($4h$ instead of $3h$ in [61], h being the local grid size).

The discrete Dirac function is also used to update the field of viscosity. As in the front-tracking method presented by Unverdi and Tryggvason [55], an indicator function $I_n = I(\mathbf{x}_n, t)$ is calculated to determine if \mathbf{x}_n is located inside ($I_n = 0$) or outside ($I_n = 1$) the particle. The dynamic viscosity thus reads

$$\mu(\mathbf{x}_n, t) = \mu^{int} + (\mu^{ext} - \mu^{int}) I_n, \quad (25)$$

with μ^{int} the internal viscosity and μ^{ext} the viscosity outside the particle. $I(\mathbf{x}, t)$ is determined as the solution of a Poisson equation:

$$\Delta I = \nabla \cdot \mathbf{G}, \quad \text{with} \quad \mathbf{G}(\mathbf{x}_n) = \sum_{m=1}^M \mathbf{n}_m dS_m \bar{w}(\mathbf{X}_m, \mathbf{x}_n). \quad (26)$$

\mathbf{G} , the right hand side of the Poisson equation, can be seen as the regularization field of the membrane normal vectors (\mathbf{n}_m , $m = 1, \dots, M$). This technique ensures that the viscosity has the right value inside and outside the particle, with a smooth transition in the membrane region, where the forces are applied.

2.5 Particle volume conservation

Whatever the choice made for the discrete Dirac function, the IBM does not intrinsically conserve the volume enclosed by the flexible membrane: interpolation does not conserve the divergence-free character of the carrying fluid flow [34]. The error indeed goes to zero when the grid is refined but this issue is sometimes pointed out as a major drawback of the IBM method [35]. Volume conservation errors can also come from the time advancement of the markers position.

In the present numerical method, the markers position is corrected to ensure volume conservation. This is obtained using a Lagrange Multiplier method to calculate the smallest correction of the markers location to conserve the particle volume.

In two dimensions, the cell is represented by a set of M markers, each marker m being connected with markers $m - 1$ and $m + 1$ (markers M and 1 are connected, so that the membrane is closed; in the following, when $m = 1$, $m - 1 = M$, and when $m = M$, $m + 1 = 1$). In 2-D, the volume enclosed in the membrane is calculated as (with \mathbf{X} the matrix of size $2 \times M$ containing the markers coordinates):

$$V(\mathbf{X}) = \frac{1}{2} \sum_{m=1}^M (X_m Y_{m+1} - X_{m+1} Y_m). \quad (27)$$

Recall that X_m and Y_m are the two coordinates of marker m .

At the beginning of the calculation, the volume of the deformable particle V_0 is calculated. At the end of each time step, the coordinates of the membrane markers (\mathbf{X}) are predicted, after time advancement of the markers position. As already stated, $V(\mathbf{X}) \neq V_0$ since the interpolation does not conserve the divergence-free character of the velocity field.

The aim is then to find the smallest markers displacements ($\delta \mathbf{X}$) in norm, so that $V(\mathbf{X} + \delta \mathbf{X}) = V_0$. $\mathbf{X}^{final} = \mathbf{X} + \delta \mathbf{X}$ denotes the matrix containing the final coordinates. Introducing a Lagrange multiplier Λ , the sought correction displacements $\delta \mathbf{X}$ minimize the following cost function:

$$J_\Lambda(\delta \mathbf{X}) = \sum_{m=1}^M \left((\delta X_m)^2 + (\delta Y_m)^2 \right) + \Lambda (V(\mathbf{X}^{final}) - V_0) \quad (28)$$

First, partial derivatives of J_Λ are calculated against δX_m and δY_m . As these derivatives are zero at the minimum, one finds $\delta X_m = -\frac{\Lambda}{4} d(Y_m^{final})$ and $\delta Y_m = \frac{\Lambda}{4} d(X_m^{final})$. The following notation is used:

$$d(X_m^{final}) = (X_{m+1}^{final} - X_{m-1}^{final}) \quad \text{and} \quad d(Y_m^{final}) = (Y_{m+1}^{final} - Y_{m-1}^{final}). \quad (29)$$

The following hypothesis is made: correction displacements are small, so that $\|d(\delta X_m)\| \ll \|d(X_m^{final})\|$ and $\|d(\delta Y_m)\| \ll \|d(Y_m^{final})\|$. Thus, quantities calculated from X_m^{final} and Y_m^{final} are approximated by their counterparts from X_m and Y_m , which yields:

$$\delta X_m = -\frac{\Lambda}{4} d(Y_m) \quad \text{and} \quad \delta Y_m = \frac{\Lambda}{4} d(X_m). \quad (30)$$

Interestingly, the normal vector to the membrane at marker m is roughly aligned with the vector $(d(Y_m), -d(X_m))^t$ (depending on the algorithm used to calculate the normal). One recovers the intuitive statement that points have to be moved along the normal direction to make the volume of the particle change. Such a reasoning is also made by [63] in their volume-conservation algorithm. However, they enforce displacement in the normal direction, calculated with a second-order polynomial approximation. Here, δX_m and δY_m are found to be optimal given the formula used to calculate the volume (Eq. 27), independently of the normal calculation.

When J_Λ is minimum, $\frac{\partial J_\Lambda}{\partial \Lambda} = 0$, which means that $V(\mathbf{X}^{final}) = V_0$. Using Eq. (27) and Eqs. (30), the following second-order polynomial equation in Λ is obtained: $A\Lambda^2 + B\Lambda + C = 0$, with:

$$A = \frac{1}{32} \sum_{m=1}^M (-d(X_{m+1}) d(Y_m) + d(X_m) d(Y_{m+1})) \quad (31)$$

$$B = \frac{1}{8} \sum_{m=1}^M (-Y_{m+1} d(Y_m) + x_i d(X_{m+1}) - X_{m+1} d(X_m) + Y_m d(Y_{m+1})) \quad (32)$$

$$C = V(\mathbf{X}) - V_0 \quad (33)$$

The smallest solution (in absolute value) Λ_{min} of equation $A\Lambda^2 + B\Lambda + C = 0$ is calculated and Eqs. (30) are used to move the markers. One thus obtains a volume-correction algorithm with negligible computational cost. In the reported calculations, one iteration is sufficient to maintain relative volume variations of the particles at less than 10^{-15} . Note that Li *et al.* [63] iterate their volume-conserving algorithm several times to obtain the desired volume conservation. This probably comes from the lack of consistency between their interface normal calculation and their volume-conservation algorithm. A similar treatment can be applied in 3-D, adapting the formula to calculate the volume. A third-order polynomial equation has then to be solved to determine the smallest value of the Lagrange multiplier.

Another method to conserve the volume consists in slightly modifying the force balance on the markers by adding a uniform normal force on the markers. The force is proportional to the volume variation and to a penalty parameter [64, 65]. The technique used here has the advantage of imposing the volume conservation more strictly and does not involve any penalty parameter. It can be considered as a projection correcting volume conservation errors related to the interpolation and to the time advancement of the markers position.

3 VALIDATIONS

This section is dedicated to the validation of the FSI solver, named YALES2BIO². As the present numerical method is not standard, validation is very detailed to prove the quality of the results. Validation notably involves test cases related to vesicles (*ie* particles with strong resistance to stretching and bending resistance) and capsules (*ie* particles with elastic membranes without bending resistance).

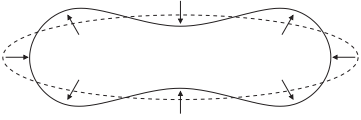
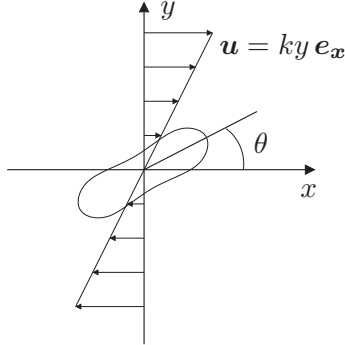
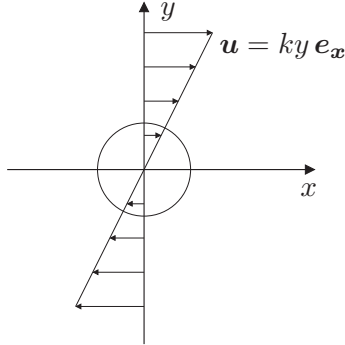
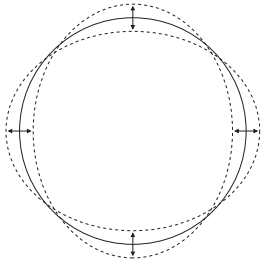
Test case (section)	Diagram	Elements tested	Sources of reference data
Equilibrium shapes of vesicles (§ 3.2)		Forces calculation, FSI coupling	Equilibrium shapes generated by minimization of surface energy
Vesicles in shear flow (§ 3.3)		FSI coupling, variable viscosity, Bending force	Veerapaneni <i>et al.</i> [14], Ghigliotti <i>et al.</i> [12] and Beaucourt <i>et al.</i> [29]
Circular capsules in shear flow (§ 3.4)		Elastic force, Capsule under Stokes flow	Breyiannis & Pozrikidis [49] and Wolfenden & Blyth [11]
Damped oscillations of capsules (§ 3.5)		Elastic force, evolution in a non-zero Reynolds number flow	Lee and Leveque [35], Cottet <i>et al.</i> [30, 66] and Martins Afonso <i>et al.</i> [67]

Table 1
Summary of the validation test cases presented in section 3.

3.1 Overview of the validation test cases

Table 1 summarizes the different test cases reported in this validation section. Several test cases are presented to be able to validate the different elements composing the FSI solver. All test cases are performed using the volume-conserving algorithm, to guarantee perfect conservation of the fluid mass enclosed in the membrane.

² <http://www.math.univ-montp2.fr/~yales2bio/>

Non-dimensional numbers used to define the geometry and the properties of the membrane and the flow field are now presented. The geometry of a vesicle is often characterized with only one parameter, the reduced volume, built from the vesicle volume and membrane area. In 2-D, the reduced area ϑ is defined as $\vartheta = 4\pi \mathcal{A} / \mathcal{P}^2$, with \mathcal{A} and \mathcal{P} the particle area and the membrane perimeter, respectively. $\mathcal{R} = \mathcal{P} / 2\pi$ is an equivalent radius of the particle. The situation is different for capsules, for which the membrane can stretch. \mathcal{R} defines the equivalent radius of the reference geometry for the capsule. For example, for a circular capsule of radius a , $\mathcal{R} = a$, whatever the deformation undergone by the capsule.

The viscosity of the internal and external fluids may differ. This is characterized by the internal-to-external viscosity ratio $\lambda = \frac{\mu^{int}}{\mu^{ext}}$.

For particles under flow, several non-dimensional numbers are used to compare the different forces involved. The Reynolds number, comparing the inertial effects to the viscous effects is defined as:

$$Re = \frac{\rho k \mathcal{R}^2}{\mu^{ext}}, \quad (34)$$

with k a characteristic shear rate. Two non-dimensional numbers, equivalent to the capillary numbers, are constructed to compare the viscous forces to the membranes elastic and bending forces, Ca and χ , respectively:

$$Ca = \frac{\mu^{ext} k \mathcal{R}}{E_e} \quad \text{and} \quad \chi = \frac{\mu^{ext} k \mathcal{R}^3}{E_b}. \quad (35)$$

Finally, β is defined to compare the bending forces and the elastic forces. β characterizes the membrane mechanics and is thus independent of the flow:

$$\beta = \frac{Ca}{\chi} = \frac{E_b}{E_e \mathcal{R}^2}. \quad (36)$$

The lower the β ratio, the closer the particle to a vesicle (elastic forces ensuring the incompressibility of the membrane).

3.2 Equilibrium shapes of vesicles

The first test case is an equilibrium test case. It is inspired by the test case of vesicle equilibrium shapes [68], used for validation of vesicle solvers [15, 29]. For vesicles, the membrane surface is incompressible. When modeled by the Helfrich energy (Eq. 4), the bending resistance makes a vesicle of arbitrary initial shape reach an equilibrium state which depends on ϑ , the reduced volume (reduced surface in 2-D). In the present validation test case, the incompressibility is not strictly enforced. Thus, it will be shown that the final equilibrium shape also depends on β (Eq. 36).

In the present validation work, particles of initially elliptic shapes (of long radius a and small radius b) are placed in a fluid at rest and evolve in time due to resistance to bending. The

reference unstressed shape (necessary for elastic force calculations) is the initial elliptic shape. Three cases are reported here. The non-dimensional parameters of the cases considered are gathered in Table 2. Two values of the reduced volume are considered, $\vartheta = 0.84$ and 0.54 . For $\vartheta = 0.54$, two values of β are used. In case 2, $\beta = 3 \times 10^{-4}$, so that the perimeter is almost conserved (as in case 1). In case 3, β is 100 times larger, so that changes in the particle perimeter are expected. The fluid viscosity only impacts the temporal evolution of the membrane shape, not its equilibrium shape. Note that despite the fact that the solution of the problem only involves the membrane, the fluid-structure coupling is tested in this case, as the membrane moves by action of the fluid. The membrane is discretized with 128 markers. The fluid flow is solved over a triangular unstructured grid of typical mesh size $0.04 a$. The grid size is of the same order as the spacing between membrane markers, as usually advised for IBM [61].

Parameter	a/b	ϑ	$E_b/E_e \mathcal{R}^2$
Case 1	2	0.84	3×10^{-4}
Case 2	4	0.54	3×10^{-4}
Case 3	4	0.54	3×10^{-2}

Table 2

Parameter values for the particle equilibrium shape test case.

In order to validate the results, shape optimization is performed to find the vesicle shape that minimizes the total energy $\mathcal{E} = \mathcal{E}_e + \mathcal{E}_b$ (see Eqs. 3 and 4), discretized as in § 2.3. Strain in the membrane and curvature are calculated as in the fluid-structure solver, so that the comparison is used to validate 1) the fluid-structure coupling and 2) the force calculation. Results are displayed in Fig. 4, which shows that equilibrium shapes are perfectly recovered in the simulations. The shapes resemble to that of 2-D vesicles when β is small (cases 1 and 2). Shapes do not depend on the initial area ratio only, but also on the membrane mechanics, as shown by the results for case 3 (Fig. 4b). Variations of the perimeter compared to the initial shape is less than 0.05 % in cases 1 and 2, but reaches 4% in case 3.

3.3 Vesicles in shear flow

The vesicle-like particles are now considered under flow, presenting the classical test case of the vesicle in a simple infinite shear flow. The limit for zero Reynolds number is considered here, to validate against BIM results. The behavior of 2-D vesicles in a simple infinite shear flow at zero Reynolds is well known: it depends mainly on the reduced volume and on the internal-to-external viscosity ratio λ [12, 14, 29]. For low viscosity ratios, the vesicle tank-treads while keeping a fixed orientation angle with the flow direction. The angle depends on the reduced volume, the vesicle being more aligned with the flow for low values of ϑ . The effect of the bending modulus on the results exists but is small [14]. When λ increases, the vesicle also aligns more and more with the flow, until the transition to tumbling is observed.

Several cases are computed to recover the dependence of the results to ϑ and λ . Simulations are performed at $Re = 0.001$, which is often considered as a good approximation of the Stokes

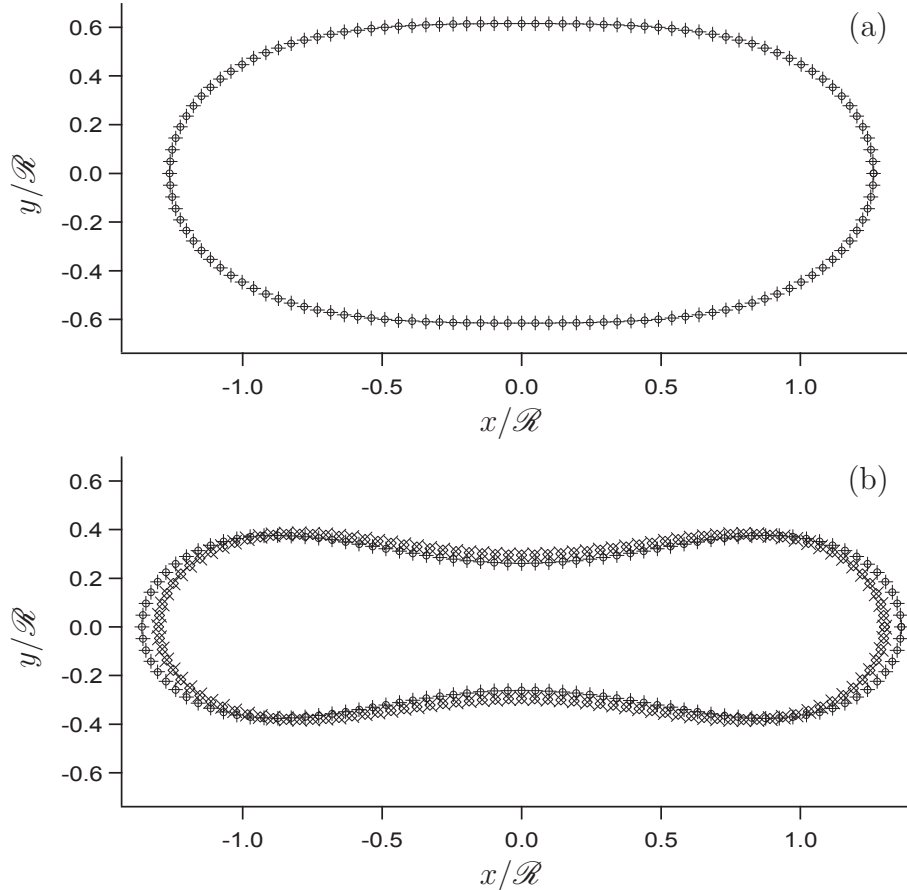


Fig. 4. Equilibrium shapes for two-dimensional particles (see Table 2) for case 1, $a/b = 2$ (a) and cases 2 and 3, $a/b = 4$ (b). Comparison between IBM results (+, \times) and shape optimization results (\circ , \diamond). Two values of β are computed, $\beta = 3 \times 10^{-4}$ (+ and \circ symbols) for cases 1 and 2 and $\beta = 3 \times 10^{-2}$ (\times and \diamond symbols) for case 3.

regime [69] (additional results on the effect of the Reynolds number are presented for the case of a capsule in shear flow). First, the viscosity ratio between the internal and external fluids is set to $\lambda = 1$ and ϑ is varied between 0.5 and 0.9.

The numerical domain and the grid are presented in Fig. 5. The numerical domain is a square box of size $100\mathcal{R} \times 100\mathcal{R}$, \mathcal{R} being the equivalent radius of the vesicle. In preliminary calculations, differences of less than 0.05 % in the equilibrium angle of tank-treading particles have been observed when changing the size of the domain from $50\mathcal{R} \times 50\mathcal{R}$ to $100\mathcal{R} \times 100\mathcal{R}$. The size of the numerical domain is sufficient to avoid the spurious effect of confinement on the results. Periodic boundary conditions are imposed in the x direction. Walls with tangential motion are used at $y = \pm 50\mathcal{R}$ to maintain a shear flow defined as $\mathbf{u}(x, y) = ky\mathbf{e}_x$ (Fig. 5, left). The domain is very large compared to the size of the particle. This can be done with a moderate extra computational cost, as the unstructured method enables the use of coarse mesh far from the vesicle (the typical mesh size near the boundaries is $4\mathcal{R}$, as shown in the Fig. 5, left). The grid size at the center of the domain (where the vesicle flows) is set to $\mathcal{R}/16$. For all the computed cases, the vesicle membrane is discretized with 128 markers (as shown

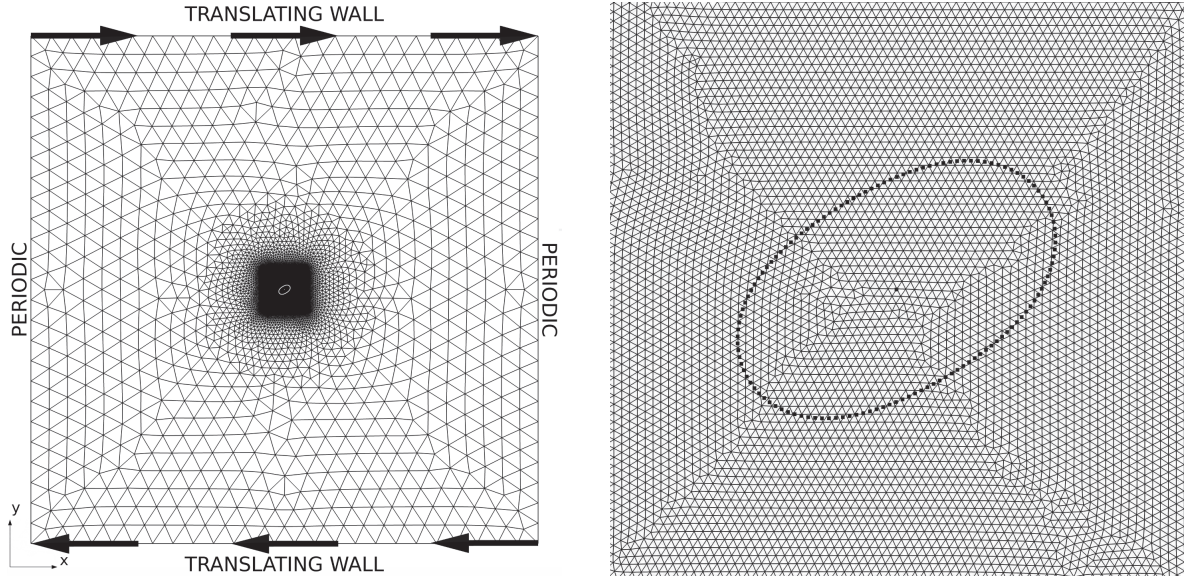


Fig. 5. Numerical domain and grid for the cases of vesicles in simple shear flow. Left: complete domain (with the boundary conditions) is shown. Right: zoom on the center of the domain; view of the fluid grid and membrane markers (black dots).

in Fig. 5, right), initially equally spaced.

The vesicles are initially elliptic, as in the former test case. Their equivalent radius \mathcal{R} is kept constant, while changing their aspect ratio to obtain different values of ϑ . In the present simulations, $Ca = 10^{-5}$, so that the viscous forces do not overcome the elastic resistance. Two values of χ are considered, 1 and 100, so that β remains very small: the difference with a truly incompressible vesicle is minimal. Indeed, relative variations of the perimeter are less than 10^{-5} for $\chi = 1.0$ and less than 10^{-7} for $\chi = 100.0$. Whatever the case, the relative local area change is never larger than 10^{-4} .

After a transient phase, vesicles reach a stable state, with constant inclination θ with respect to the flow, the membrane tank-treading around the internal fluid. Fig. 5(right) shows the final orientation for a vesicle of reduced area 0.9. Equilibrium angles are calculated by assessing the tensor of the moments of inertia [49]. The comparison of equilibrium angles as a function of the reduced volume is shown in Fig. 6. Very good agreement is obtained with existing boundary integral results [14]. The present simulations reproduce well the small yet visible dependence of the results on χ .

The ability to compute configurations with different viscosity values inside and outside the particles is now verified. For $\chi = 1$ and $\vartheta = 0.9$, for a given external viscosity, the internal viscosity is varied. In the tank treading regime, the equilibrium angle decreases with the internal-to-external viscosity ratio λ . Excellent comparisons are obtained (Fig. 7) with the numerical results for two-dimensional vesicles by Ghigliotti *et al.* [12] with BIM and Beaucourt *et al.* [29] with an advected field method. Tank-treading-to-tumbling transition is found for a critical value of λ between 5.5 and 5.6, as in [12]. Note that the smooth representation of the viscosity discontinuity through the membrane does not introduce any measurable error in

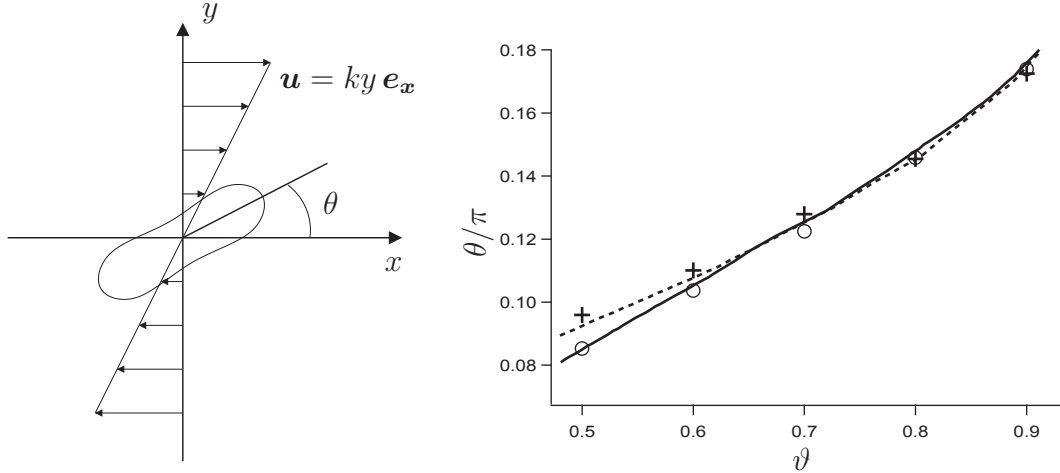


Fig. 6. Tank-treading vesicle in simple shear flow: equilibrium angle θ as a function of ϑ for $\chi = 1$ (\circ) and $\chi = 100$ ($+$). The viscosity ratio is kept at 1.0. Comparison with the numerical results of Veerapaneni *et al.* at $\chi = 1$ (—) and $\chi = 100$ (---) is provided [14].

these results.

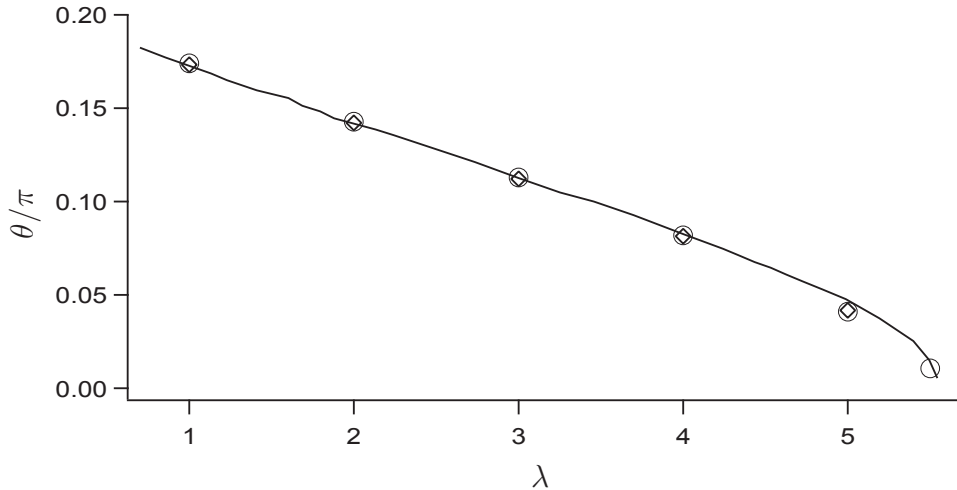


Fig. 7. Vesicle in simple shear flow: equilibrium angle as a function of the internal-to-external viscosity ratio λ for $\chi = 1$ and $\vartheta = 0.9$. Comparison of the present simulations (\circ) with the numerical results from [12] (—) and [29] (\diamond).

3.4 Capsules in shear flow

While studies of two-dimensional vesicles are not that rare, even using a boundary-integral method, reference results for elastic capsules in two dimensions are scarce. BIM simulations are rarely performed in 2-D: the computational cost being limited, authors directly published 3-D results. Two-dimensional capsules were considered in other studies, notably as a step in the development of numerical methods [30, 35, 38, 70, 71]. However, they often rely on diffuse interface methods (as the immersed boundary method), that involve more sources of approximations than BIM. As a consequence, we strongly believe that careful validations should

be performed as much as possible against BIM results, when available. In order to validate their BIM solver to study capsules in bifurcations, Woolfenden & Blyth [11, 50] used the case of an initially circular capsule in a simple shear flow from Breyiannis & Pozrikidis [49]. This test case is reproduced here. Cases with identical viscosity inside and outside the capsule are shown. In the zero Reynolds number limit, the results only depend on Ca . When the initially circular capsule is deposited in the flow, it is progressively deformed by the flow. Beyond a certain deformation, the capsule elastic forces become large enough to resist the viscous forces. The capsule stops deforming and undergoes the tank-treading motion also visible for vesicles. Elastic forces decrease with Ca : at high values of Ca , capsules are very elongated and tend to align more with the main flow.

An initially circular elastic capsule, showing no bending resistance, is deposited in an infinite shear flow $\mathbf{u}(x, y) = ky \mathbf{e}_x$. The initial capsule radius is denoted by \mathcal{R} . The initial shape is considered as the reference shape for the calculation of the stresses. The same fluid is used inside and outside the capsule ($\lambda = 1$), with a given viscosity μ .

The computational domain and boundary conditions are identical to the ones used to compute vesicles in simple shear flow (see Fig. 5). The fluid domain is a square box of size $100 \mathcal{R} \times 100 \mathcal{R}$ with periodic boundary conditions in the flow direction and translating walls in the other direction to maintain the shear flow. As already stated in the former section, this size of numerical domain is sufficient to avoid confinement effects. Grid size is set to $\mathcal{R}/16$ at the center of the numerical domain, where the capsule is deposited and of order of $4\mathcal{R}$ near the boundaries. The capsule membrane is discretized with equally-spaced 128 markers. As reference results are generated in the Stokes regime, the Reynolds number is set to $Re = 0.001$ in the present simulations (differences with the zero Reynolds number limit are less than 0.3 %, see § 4.1).

Figure 8 shows the time evolution of the deformation parameter D and the orientation θ of the capsule with the x axis. These quantities are calculated by computing the tensor of the moments of inertia [49], and identify the result with the one of an ellipse of major radius A and minor radius B and of orientation θ . D is defined as $D = (A - B)/(A + B)$. Time is made dimensionless using the characteristic time k^{-1} .

The present results match the data from [49]. The behavior in time is well reproduced, and capsules reach the correct deformation and orientation at convergence. Maximum error of 0.003 rad (3%) is observed for the final orientation of the capsule at $Ca = 0.125$.

3.5 Damped oscillations of an elastic capsule

This test case is based on numerous publications about numerical methods for flexible surfaces in flows [30, 35, 70–72]. It focuses on the relaxation of an elastic capsule. The test case configuration is summarized in Fig. 9: an initially stretched elliptic membrane, enclosing an incompressible fluid, is placed in a square fluid domain, closed by walls. The characteristics of the incompressible fluid are identical inside and outside the cell. The zero-stress configuration

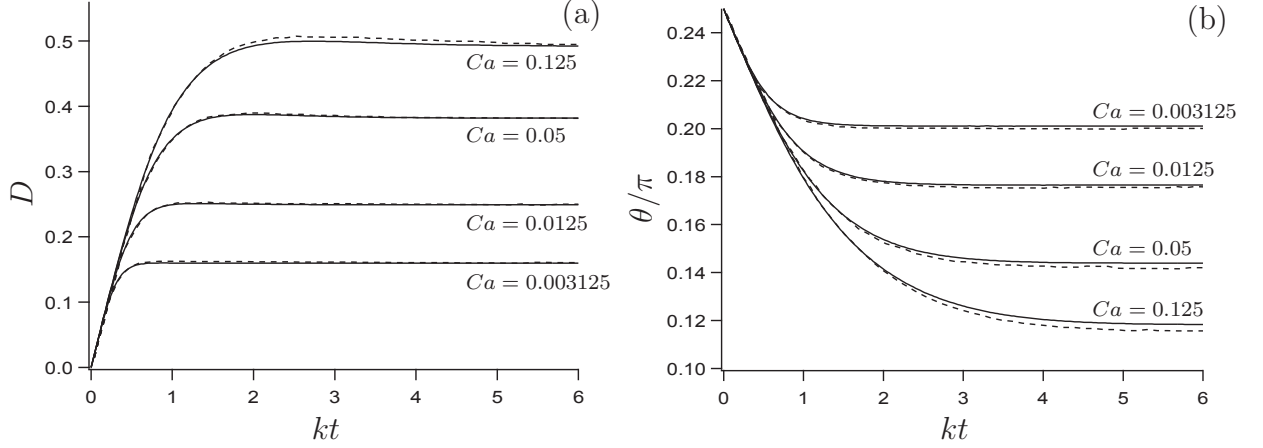


Fig. 8. Tank-treading capsule in simple shear flow: time-evolution of the deformation parameter D (a) and equilibrium angle θ/π (b) for different values of Ca , compared with existing BIM results [49] (----). Note that Fig. 2 of their paper [49] has typos in the values of Ca : results at $Ca = 0.05$ are erroneously labelled $Ca = 0.04$ in their paper.

is a circular membrane, of radius R_0 . The initial configuration is an ellipse of radii R_0 and $R_1 = 1.5 R_0$. The membrane evolves to minimize its surface (perimeter in 2-D) with prescribed volume: given the incompressible nature of the fluid, the equilibrium configuration is a circle of radius $R_f = \sqrt{R_0 R_1}$. This equilibrium configuration is pressurized.

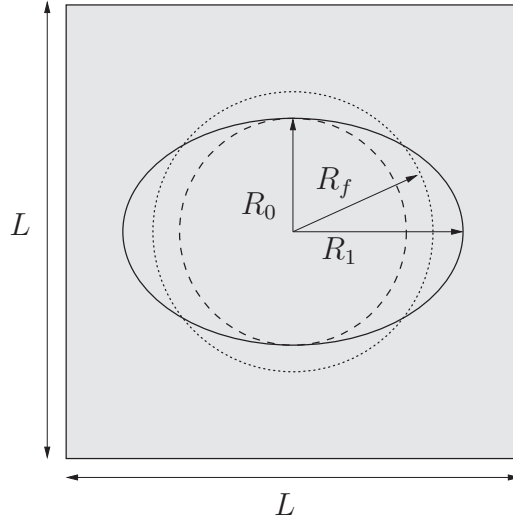


Fig. 9. Two-dimensional elastic ellipse membrane test case: the grey zone is the fluid domain. The solid line is the initial membrane location. The dashed line is the resting configuration of the unstressed membrane. The dotted line is the final equilibrium configuration.

Depending on the elastic modulus of the membrane and the fluid dynamic viscosity μ , the membrane oscillates or not before reaching the equilibrium state. More precisely, for low viscosity values, the capsule oscillates and the period of the oscillations is shown to be related with an elastic characteristic time $t_e = (\rho R_0^4 / E_e (R_f - R_0))^{\frac{1}{2}}$ [67, 73]. The damping rate of the oscillation is related to the viscous time $t_D = \rho R_0^2 / \mu$. t_e / t_D is thus a ratio between elastic and viscous time scales. The larger this ratio, the less the capsule oscillates before reaching its

circular equilibrium shape. Table 3 summarizes the values of the parameters used in the calculations. These values have been chosen to allow comparison with existing results [30, 35, 66].

Parameter	L/R_0	R_1/R_0	R_f/R_0	t_e/t_D
Value	2	1.5	1.225	0.03

Table 3

Values of the input parameters for the computation of the capsule relaxation.

Simulations are run over a fluid grid of 256×256 square cells. The membrane is represented by 500 regularly spaced markers. During the simulation, the membrane conserves approximately the shape of an ellipse, with minor and major axes along the x and the y axes. Non-dimensional diameters along the x and the y axes (initially $2R_1/R_0$ and 2, respectively), are extracted from the simulations and displayed in Fig. 10 over time. Time is made dimensionless using the elastic characteristic time t_e . Results are compared with published data by Lee & Leveque (2003) [35] with an immersed interface method and by Cottet & Maitre (2006) [30] and Cottet, Maitre & Milcent (2008) [66] with a level-set method, generated over grids of similar resolution.

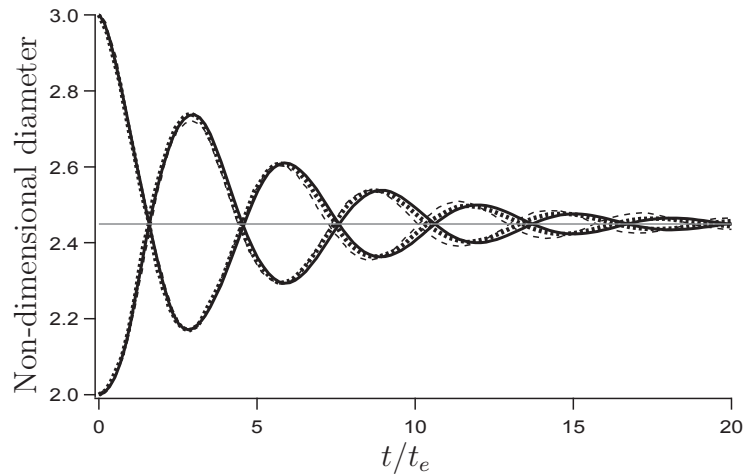


Fig. 10. Time-evolution of the diameters along the x and the y axes of an initially elliptic oscillating capsule. Comparison of the present results (—) with existing results from Lee and Leveque [35] (.....) and Cottet *et al.* [30, 66] (----). The equilibrium diameter $2R_f/R_0$ is displayed as a solid grey line.

The time evolution of the capsule diameters along the x and y directions is in good agreement with the previous results in the literature. Note in particular that the present formulation ensures perfect volume conservation along time despite the action of unsteady dilatational elastic forces.

In order to further analyze the oscillatory relaxation of 2-D capsules, Martins Afonso *et al.* [67] used normal-mode analysis to obtain semi-analytical frequency and damping rate of the oscillations of capsules. The analysis stands for small perturbations of the different azimuthal modes (the case shown in Fig. 9 corresponds to large perturbations of mode 2). As an example of the results obtained, the case of a non-confined capsule is considered, with $R_f/R_0 = 2$ and $t_e/t_D = 0.001$. The normal-mode analysis yields the complex pulsation of the proper mode of the system. Table 4 displays the complex pulsation for azimuthal modes 2 and 3,

together with the results obtained in numerical simulation (in a the small perturbation regime) using the numerical method presented in this paper. The complex pulsation values are made dimensionless using t_e^{-1} .

Case and parameter values	Normal-mode analysis	Numerical solution
Case 1 : $R_f/R_0 = 2$, $t_e/t_D = 0.001$, mode 2	$\omega t_e = 1.695 - 0.037 i$	$\omega t_e = 1.670 - 0.040 i$
Case 2 : $R_f/R_0 = 2$, $t_e/t_D = 0.001$, mode 3	$\omega t_e = 3.395 - 0.071 i$	$\omega t_e = 3.335 - 0.067 i$

Table 4

Non-dimensional complex pulsation ωt_e of proper modes of capsule relaxation in a non-confined environment. Comparison between normal-mode analysis and the present numerical method.

Table 4 illustrates the ability of the present numerical method to reproduce results from linear analysis for the oscillatory relaxation of 2-D capsules, in the limit of small perturbations. Frequency is recovered with less than a 2% error. Small yet larger differences are observed for the damping rate (8% and 5% for modes 2 and 3, respectively). A more detailed comparison of the results is presented by Martins Afonso *et al.* [67]. This very good comparison confirms the correct behavior of the present method in the test case of damped oscillations of an elastic capsule.

4 APPLICATIONS

The study of microcirculation is one of the primary objectives for developing numerical methods based on boundary integrals. Indeed, the use of relatively simple geometries and the zero-Reynolds number assumption are not restrictive in that context. The Reynolds number in capillaries is indeed of order of 10^{-3} . In addition, in the circulatory system, the shear stress seen by red blood cells remains moderate (except in some specific cases as severe stenoses [74]) so that the particle Reynolds number remains small. On the contrary, flows in medical devices are sometimes associated with very different conditions. Red blood cells may interact with manufactured elements having complex geometrical details. Moreover, the flow Reynolds number can reach 10^4 or 10^5 in ventricular assist devices and blood pumps [45]. The particle Reynolds number also gets of order of unity in devices as cytometers [75]. In order to demonstrate the performances of the present method to compute flows in non-physiological contexts, this section is dedicated to the computation of two cases: capsules under a simple shear flow for different Reynolds numbers (§ 4.1) and the entry of a capsule in a contraction, with a more complex geometry, the carrying flow being at $Re = 220$ (§ 4.2).

4.1 Effect of the Reynolds number in the case of a capsule in shear flow

The configuration of a capsule suspended in a simple shear flow, already presented in section 3.4, is used here to show the effect of the particle Reynolds number on the results. Four different Reynolds numbers are considered: $Re = 0.001$ (already shown for the comparison

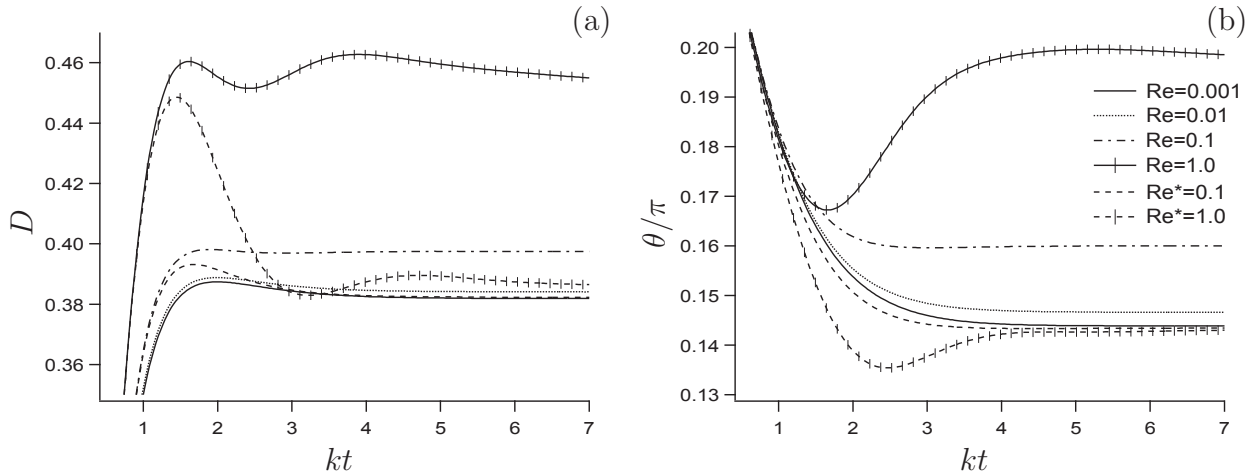


Fig. 11. Capsule in simple shear flow: time-evolution of the deformation rate (a) and inclination angle (b) at $Ca = 0.05$ for different values of the Reynolds number. The viscosity ratio is kept at 1.0. Lines labeled with Re^* are obtained from calculations with the quadratic terms of the Navier-Stokes equations turned off.

against the Stokes flow results in section 3.4), $Re = 0.01$, $Re = 0.1$ and $Re = 1.0$. The elastic capillary number is $Ca = 0.05$ for all calculations. The time-evolution of the deformation rate D and the angle with the main flow direction θ/π (defined as in Fig. 6 and Fig. 8) is shown in Fig. 11. When the Reynolds number is below one, the evolution is rather monotonic, the equilibrium values of D and θ/π decreasing with the Reynolds number. At $Re = 1.0$, oscillations are observed before reaching equilibrium. In addition, the capsule needs much more time to reach the equilibrium (typically $kt \approx 30$). Such behaviors are consistent with the results obtained in 3-D by Le *et al.* [39], for instance.

In order to quantify the importance of the contribution of the non-linear terms of the Navier-Stokes equations, they have been turned off in two computations at Reynolds numbers 0.1 and 1.0: the unsteady Stokes equations are thus solved to obtain the results labelled $Re^* = 0.1$ and $Re^* = 1.0$ in Fig. 11. The * superscript thus denotes the absence of quadratic terms (second term of the left hand side of Eq. 2) in the simulation. When the equilibrium steady state is obtained, such simulations provide the results for infinitely small values of the Reynolds number. Indeed, differences with $Re = 0.001$ are negligible: for $kt = 7$, the steady state is reached and results in the equilibrium angle differ by less than 0.3 % between $Re = 0.001$ and $Re^* = 0.1$. On the contrary, the transient behavior for these simulations is very different from the one observed at low Reynolds number. Contrary to what occurs in creeping flows, the flow does not adapt instantaneously to the membrane state: despite the absence of quadratic terms, inertia is still present due to the unsteady term. One must keep in mind that both effects are neglected when computing low-Reynolds number flows with the Stokes flow assumption (as in BIM), and that both can have an impact on the flow results. The present results also show that unsteady Stokes equations can be used to obtain quicker estimates of steady state values, but certainly not transient information.

4.2 Entry of a capsule in a sudden contraction at Reynolds number 220

In artificial devices, the flow can reach regimes that never occur in a physiological environment. Typical examples are blood pumps or ventricular assist devices, where the Reynolds number may reach 10^5 [45]. Another example is the flow in cytometers, where the fluid velocity reaches several meters per second in apertures of a few tens of microns [75]. The flow Reynolds number may reach several hundreds, in spite of the small size of the device. Typical maximal shear rate in these systems is of order of 10^6 s^{-1} . As an illustration of the flexibility of the method and its ability to deal with such non-physiological conditions, the entry of a capsule in a contraction is computed. Lengths and velocity scales are typical of industrial cytometers [75].

Figure 12 sketches the geometry of the problem. The fluid flows from left to right in a 20:1 contraction. The axes origin is located at the beginning of the contraction, on the centerline. The half-height of the narrow channel is $H = 25 \mu\text{m}$. The contraction does not have sharp edges: the entry is rounded, with a curvature radius of $5 \mu\text{m}$. It is one of the advantages of an unstructured method to be able to easily account for such geometrical details at no additional cost. A zoom near the contraction corner ($x = 0$ and $y/H = 1$) is shown in Fig. 12 to visualize the grid (a grid twice as coarse as the simulation is shown for better visualization). The flow grid contains 280,000 triangular elements. The minimum grid size is $0.35 \mu\text{m}$ around the contraction entrance. The membrane is discretized with 128 markers. The Reynolds number based on the bulk velocity $u_b = 4.4 \text{ m}\cdot\text{s}^{-1}$ and the total height of the channel is 220. An estimate of the shear rates encountered in the system is $k = u_b/H = 1.76 \cdot 10^5 \text{ s}^{-1}$.

An initially circular capsule is deposited upstream of the contraction, slightly above the centerline (its center is placed at $x = -2H$ and $y = 2H$). The initial radius of the capsule is $\mathcal{R} = 4 \mu\text{m}$. During its displacement through the contraction, it undergoes acceleration and extension along its flow direction, then shear inside the small channel. k is a good estimate of the shear rates seen by the capsules during its movement. The particle Reynolds number is thus of order of $Re = 2.7$. The fluid viscosity is the same inside and outside the particle: $\lambda = 1$. The capsule capillary number is set to $Ca = 0.0136$ to allow visible deformations while preventing excessive extensions.

Figure 13 shows the dynamics of the capsule over time, superimposed with the streamwise velocity field (in absence of the capsule). The capsule undergoes a rapid acceleration when flowing through this 20:1 contraction. Starting from its initial location (label a in Fig. 13), it first lengthens in the direction of the flow (see label b, at $t = 5.61k^{-1}$). Inside the contraction, the velocity gradient becomes negative along the trajectories, which results in a contraction of the capsule, which recovers a more circular shape ($t = 6.63k^{-1}$, label c) than at the entrance. Further in the channel, the influence of the walls are seen: the capsule enters in a zone with very small extensional motion, but with important shear. Shape now resembles the off-centerline vesicle shapes described for instance by Kaoui *et al.* [76] in an unbounded plane Poiseuille flow. Such flow is of great interest to test the numerical method, as the carrying flow presents successively important extensional, compressional and eventually shear characters.

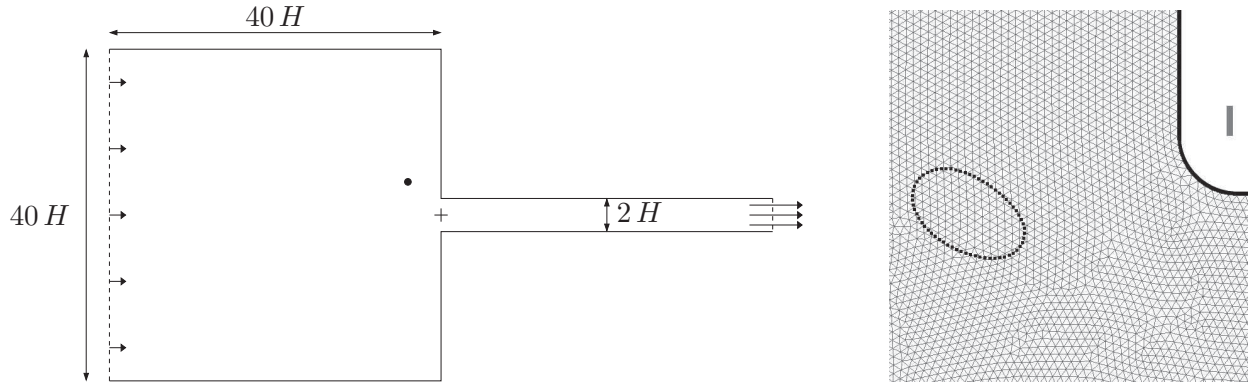


Fig. 12. Left: sketch of the contraction configuration (not to scale). + marks the axes origin and the dot marks the initial location of the capsule. Flow goes from left (inflow) to right (outflow). Solid lines denote non-slipping walls. Right: zoom near the contraction entrance with fluid triangular grid and view of the membrane markers (at an arbitrary time during the simulation). Every second point is shown for better visualization of the grid. The length of the thick grey line in the top right corner is $H/10$.

Note that the streamline issuing from $x = -2H$ and $y = 2H$ and calculated from the carrying flow (before the capsule deposit) is also displayed as a black line in Fig. 13. It shows that the capsule essentially follows the trajectory that tracers would have in the extension-compression zone (upstream of $x = 2H$). Lateral migration is observed once the capsule lies in a zone of substantial shear ($x > 2H$): the center of mass of the capsule progressively moves away from the streamline (see label d, at $t = 8.67k^{-1}$).

5 CONCLUSION

A numerical method for simulations of the dynamics of deformable objects constituted by a liquid droplet enclosed in a flexible structure is presented. The approach is based on the immersed boundary method adapted to unstructured grids with the reproducing kernel particle method. The specificity of the method is to enable calculations at high Reynolds number, in complex geometries using body-fitted unstructured grids. The unstructured IBM is supplemented with a volume-conservation algorithm that ensures that interpolation of the velocity field to the membrane markers in the IBM does not spoil the divergence-free character of the velocity field.

The method demonstrates its ability to handle both low and high Reynolds number flows, for different properties of the immersed membrane. Specific attention is devoted to the careful validation of the numerical method. In this regard, a series of validation test cases are presented, gathering capsules and vesicles configurations. We believe that this collection of 2-D test cases constitute a good *quantitative* validation process for 2-D methods, ruling out the use of *qualitative* comparisons with 3-D experiments or simulations, sometimes used as validation in method papers. Robustness of the numerical method in a complex flow involving a sequence of extension-compression-shear of the deformable particle is also shown.

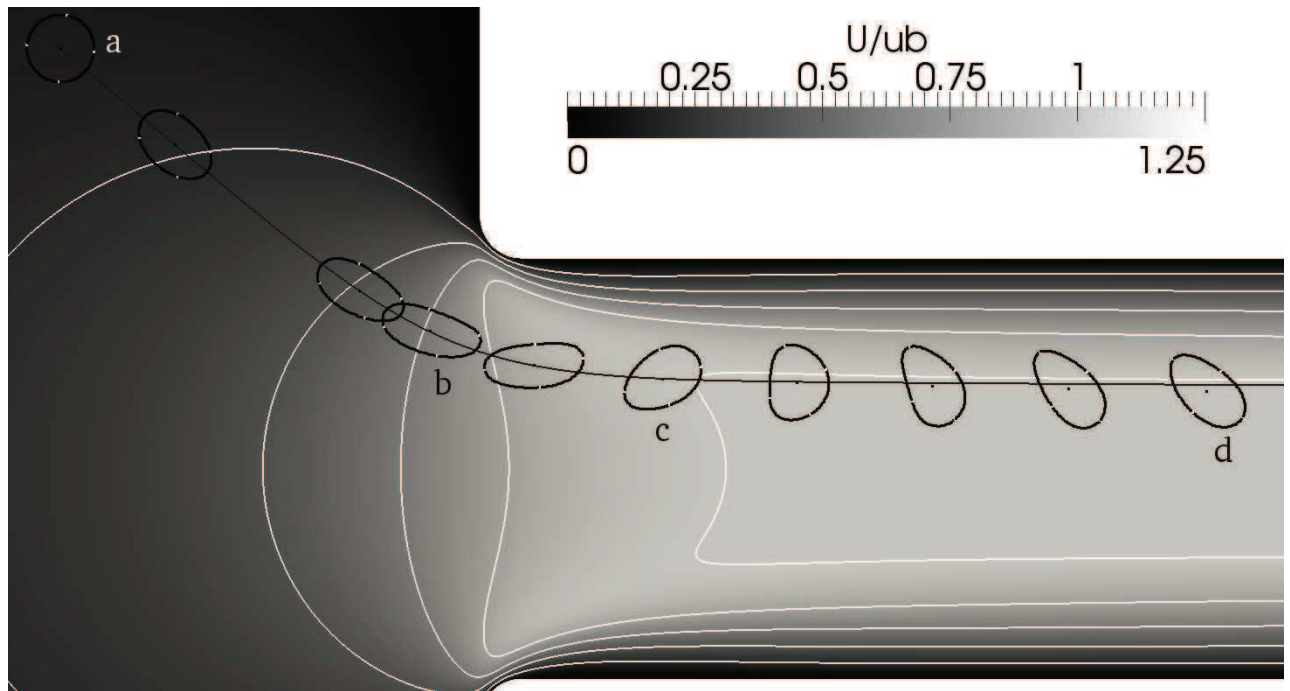


Fig. 13. Sequence of membrane locations superimposed over the carrying flow (the x component of velocity, U , is displayed: white iso-lines of U/u_b are shown each 0.25, from 0.25 to 1.25). The black line is the streamline (computed from the carrying flow) issuing from the initial location of the capsule center of mass. Membrane markers are shown at $t = 0$, $t = 2.55k^{-1}$ and $t = 5.10k^{-1}$, then every $t = 0.51k^{-1}$. The center of mass is displayed at the center of the particle. Labels a to d mark the membrane position for $kt = 0$, 5.61, 6.63 and 8.67, respectively.

ACKNOWLEDGMENTS

The authors acknowledge support from ANR, through the project FORCE (Flows Of Red blood CELls), in which the YALES2BIO solver has been developed. The authors also acknowledge the support of OSEO, via the DAT@DIAG project. Vincent Moureau, Vanessa Lleras and Marco Martins Afonso are thanked for helpful discussions. The authors also thank the NUMEV Labex for support.

References

- [1] C. Misbah, Vesicles, capsules and red blood cells under flow, *Journal of Physics: Conference Series* 392 (2012) 012005.
- [2] C. Pozrikidis, *Modeling and Simulation of Capsules and Biological Cells*, Boca Raton: Chapman & Hall/CRC, 2003.
- [3] C. Pozrikidis, *Computational Hydrodynamics of Capsules and Biological Cells*, Boca Raton: Chapman & Hall/CRC, 2010.
- [4] D. Barthès-Biesel, Capsule motion in flow: Deformation and membrane buckling, *Comptes Rendus Physique* 10 (2009) 764–774.

- [5] D. Barthès-Biesel, Modeling the motion of capsules in flow, *Current Opinion in Colloid & Interface Science* 16 (2011) 3–12.
- [6] C. Pozrikidis, *Boundary Integral and Singularity Methods for Linearized Viscous Flow*, Cambridge University Press., 1992.
- [7] S. Ramanujan, C. Pozrikidis, Deformation of liquid capsules enclosed by elastic membranes in simple shear flow: large deformations and the effect of fluid viscosities, *Journal of Fluid Mechanics* 361 (1998) 117–143.
- [8] E. Lac, A. Morel, D. Barthès-Biesel, Hydrodynamic interaction between two identical capsules in simple shear flow, *Journal of Fluid Mechanics* 573 (2007) 149–169.
- [9] W. R. Dodson III, P. Dimitrakopoulos, Dynamics of strain-hardening and strain-softening capsules in strong planar extensional flows via an interfacial spectral boundary element algorithm for elastic membranes, *Journal of Fluid Mechanics* 641 (2009) 263–296.
- [10] J. Walter, A.-V. Salsac, D. Barthès-Biesel, P. Le Tallec, Coupling of finite element and boundary integral methods for a capsule in a Stokes flow, *International Journal for Numerical Methods in Engineering* 83 (2010) 829–850.
- [11] H. C. Woolfenden, M. G. Blyth, Motion of a two-dimensional elastic capsule in a branching channel flow, *Journal of Fluid Mechanics* 669 (2011) 3–31.
- [12] G. Ghigliotti, T. Biben, C. Misbah, Rheology of a dilute two-dimensional suspension of vesicles, *Journal of Fluid Mechanics* 653 (2010) 489–518.
- [13] T. Biben, A. Farutin, C. Misbah, Three-dimensional vesicles under shear flow: Numerical study of dynamics and phase diagram, *Physical Review E* 83 (2011).
- [14] S. K. Veerapaneni, D. Gueyffier, D. Zorin, G. Biros, A boundary integral method for simulating the dynamics of inextensible vesicles suspended in a viscous fluid in 2D, *Journal of Computational Physics* 228 (2009) 2334–2353.
- [15] G. Boedec, M. Leonetti, M. Jaeger, 3D vesicle dynamics simulations with a linearly triangulated surface, *Journal of Computational Physics* 230 (2011) 1020–1034.
- [16] H. Zhao, E. S. Shaqfeh, The dynamics of a vesicle in simple shear flow, *Journal of Fluid Mechanics* 674 (2011) 578–604.
- [17] C. Pozrikidis, Numerical simulation of the flow-induced deformation of red blood cells, *Annals of Biomedical Engineering* 31 (2003) 1194–1205.
- [18] H. Zhao, A. H. G. Isfahani, L. N. Olson, J. B. Freund, A spectral boundary integral method for flowing blood cells, *Journal of Computational Physics* 229 (2010) 3726–3744.
- [19] Z. Peng, R. J. Asaro, Q. Zhu, Multiscale modelling of erythrocytes in Stokes flow, *Journal of Fluid Mechanics* 686 (2011) 299–337.
- [20] H. Noguchi, G. Gompper, Swinging and tumbling of fluid vesicles in shear flow, *Physical Review Letters* 98 (2007) 128103.
- [21] J. L. McWhirter, H. Noguchi, G. Gompper, Flow-induced clustering and alignment of vesicles and red blood cells in microcapillaries, *Proceedings of the National Academy of Sciences USA* 106 (2009) 6039–6043.
- [22] W. Pan, B. Caswell, G. E. Karniadakis, A low-dimensional model for the red blood cell, *Soft Matter* 6 (2010) 4366–4376.
- [23] D. A. Fedosov, B. Caswell, G. E. Karniadakis, A multiscale red blood cell model with accurate mechanics, rheology, and dynamics, *Biophysical Journal* 98 (2010) 2215–2225.
- [24] Y. Sui, Y. T. Chew, P. Roy, Y. P. Cheng, H. T. Low, Dynamic motion of red blood cells in simple shear flow, *Physics of Fluids* 20 (2008).

- [25] M. MacMeccan, J. R. Clausen, G. P. Neitzel, C. K. Aidun, Simulating deformable particle suspensions using a coupled lattice-Boltzmann and finite-element method, *Journal of Fluid Mechanics* 618 (2009) 13–39.
- [26] T. Gao, H. H. Hu, Deformation of elastic particles in viscous shear flow, *Journal of Computational Physics* 228 (2009) 2132–2151.
- [27] C. Bui, V. Lleras, O. Pantz, Dynamics of red blood cells in 2D, *ESAIM: Proceedings* 28 (2009) 182–194.
- [28] T. Biben, C. Misbah, Tumbling of vesicles under shear flow within an advected-field approach, *Physical Review E* 67 (2003).
- [29] J. Beaucourt, F. Rioual, T. Séon, T. Biben, C. Misbah, Steady to unsteady dynamics of a vesicle in a flow, *Physical Review E* 69 (2004).
- [30] G.-H. Cottet, E. Maitre, A level set method for fluid-structure interactions with immersed surfaces, *Mathematical Models and Methods in Applied Science* 16 (2006) 415–438.
- [31] E. Maitre, T. Milcent, G.-H. Cottet, A. Raoult, Y. Usson, Applications of level set methods in computational biophysics, *Mathematical and Computing Modeling* 49 (2009) 2161–2169.
- [32] D. Salac, M. Miksis, A level set projection model of lipid vesicles in general flows, *Journal of Computational Physics* 230 (2011) 8192–8215.
- [33] A. Laadhari, P. Saramito, C. Misbah, Vesicle tumbling inhibited by inertia, *Physics of Fluids* 24 (2012) 031901.
- [34] C. S. Peskin, The immersed boundary method, *Acta Numerica* 11 (2002) 479–517.
- [35] L. Lee, R. J. Leveque, An immersed interface method for incompressible Navier–Stokes equations, *SIAM Journal of Scientific Computing* 25 (2003) 832–856.
- [36] X. Wang, W. K. Liu, Extended immersed boundary method using FEM and RKPM, *Computer Methods in Applied Mechanics and Engineering* 193 (2004) 1305–1321.
- [37] Y. Kim, M.-C. Lai, Simulating the dynamics of inextensible vesicles by the penalty immersed boundary method, *Journal of Computational Physics* 229 (2010) 4840–4853.
- [38] D.-V. Le, B. C. Khoo, J. Peraire, An immersed interface method for viscous incompressible flows involving rigid and flexible boundaries, *Journal of Computational Physics* 220 (2006) 109–138.
- [39] D.-V. Le, Large deformation of liquid capsules enclosed by thin shells immersed in the fluid, *Journal of Computational Physics* 229 (2010) 4097–4116.
- [40] D.-V. Le, K.-H. Chiam, Hydrodynamic interaction between two nonspherical capsules in shear flow, *Physical Review E* 84 (2011).
- [41] P. Bagchi, R. M. Kalluri, Dynamic rheology of a dilute suspension of elastic capsules: effect of capsule tank-treading, swinging and tumbling, *Journal of Fluid Mechanics* 669 (2011).
- [42] A. Z. K. Yazdani, P. Bagchi, Influence of membrane viscosity on capsule dynamics in shear flow, *Journal of Fluid Mechanics* 718 (2013) 569–595.
- [43] M. H.-Y. Tan, D.-V. Le, K.-H. Chiam, Hydrodynamic diffusion of a suspension of elastic capsules in bounded simple shear flow, *Soft Matter* 8 (2012) 2243–2251.
- [44] A. Z. K. Yazdani, R. M. Kalluri, P. Bagchi, Tank-treading and tumbling frequencies of capsules and red blood cells, *Physical Review E* 83 (2011).
- [45] K. H. Fraser, M. E. Tskain, B. P. Griffith, Z. J. Wu, The use of computational fluid dynamics in the development of ventricular assist devices, *Medical Engineering and Physics*

- 33 (2011) 263–280.
- [46] L. Zhang, A. Gerstenberger, X. Wang, W. K. Liu, Immersed finite element method, *Computer Methods in Applied Mechanics and Engineering* 193 (2004) 2051–2067.
 - [47] Y. Liu, W. K. Liu, Rheology of red blood cell aggregation by computer simulation, *Journal of Computational Physics* 220 (2006) 139–154.
 - [48] W. K. Liu, D.-W. Kim, S. Tang, Mathematical foundations of the immersed finite element method, *Computational Mechanics* 39 (2007) 211–222.
 - [49] G. Breyiannis, C. Pozrikidis, Simple shear flow of suspensions of elastic capsules, *Theoretical and Computational Fluid Dynamics* 13 (2000) 327–347.
 - [50] H. C. Woolfenden, The transit of an elastic capsule through a bifurcation, Ph.D. thesis, School of Mathematics of the University of East Anglia, 2010.
 - [51] W. Helfrich, Elastic properties of lipid bilayers: Theory and possible experiments, *Z. Naturforsch* 28 c (1973) 693–703.
 - [52] O. Y. Zhong-can, W. Helfrich, Bending energy of vesicle membranes: General expressions for the first, second, and third variation of the shape energy and applications to spheres and cylinders, *Physical Review A* 39 (1989) 5280–5288.
 - [53] C. S. Peskin, Numerical analysis of blood flow in the heart, *Journal of Computational Physics* 25 (1977) 220–252.
 - [54] A. M. Roma, C. S. Peskin, M. J. Berger, An adaptive version of the immersed boundary method, *Journal of Computational Physics* 153 (1999) 509–534.
 - [55] S. O. Unverdi, G. Tryggvason, A front-tracking method for viscous, incompressible, multi-fluid flows, *Journal of Computational Physics* 100 (1992) 25–37.
 - [56] P. Bagchi, P. C. Johnson, A. S. Popel, Computational fluid dynamic simulation of aggregation of deformable cells in a shear flow, *Journal of Biomechanical Engineering* 127 (2005) 1070–1080.
 - [57] M. Malandin, N. Maheu, V. Moureau, Optimization of the deflated conjugate gradient algorithm for the solving of elliptic equations on massively parallel machines, *Journal of Computational Physics* 238 (2013) 32–47.
 - [58] V. Moureau, P. Domingo, L. Vervisch, Design of a massively parallel CFD code for complex geometries, *Comptes Rendus Mécanique* 339 (2011) 141–148.
 - [59] V. Moureau, P. Domingo, L. Vervisch, From large-eddy simulation to direct numerical simulation of a lean premixed swirl flame: Filtered laminar flame-PDF modelling, *Computers and Fluids* 158 (2011) 1340–1357.
 - [60] A. Chorin, Numerical solution of the Navier–Stokes equations, *Mathematics of Computation* 22 (1968) 745–762.
 - [61] A. Pinelli, I. Z. Naqavi, U. Piomelli, J. Favier, Immersed-boundary methods for general finite-difference and finite-volume Navier–Stokes solvers, *Journal of Computational Physics* 229 (2010) 9073–9091.
 - [62] W. K. Liu, S. Jun, Y. F. Zhang, Reproducing kernel particle methods, *International Journal for Numerical Methods in Fluids* 20 (1995) 1081–1106.
 - [63] Y. Li, E. Jung, W. Lee, H. G. Lee, J. Kim, Volume preserving immersed boundary methods for two-phase fluid flows, *International Journal for Numerical Methods in Fluids* 69 (2012) 842–858.
 - [64] Z. Peng, R. J. Asaro, Q. Zhu, Multiscale simulation of erythrocyte membranes, *Physical Review E* 81 (2010).

- [65] A. Z. K. Yazdani, P. Bagchi, Three-dimensional numerical simulation of vesicle dynamics using a front-tracking method, *Physical Review E* 85 (2012).
- [66] G.-H. Cottet, E. Maitre, T. Milcent, Eulerian formulation and level set models for incompressible fluid-structure interaction, *ESAIM: Mathematical Modelling and Numerical Analysis* 42 (2008) 471–492.
- [67] M. Martins Afonso, S. Mendez, F. Nicoud, On the damped oscillations of an elastic quasi-circular membrane in a two-dimensional incompressible fluid, submitted to *Journal of Fluid Mechanics* (2013).
- [68] U. Seifert, K. Berndl, R. Lipowsky, Shape transformations of vesicles: Phase diagram for spontaneous-curvature and bilayer-coupling models, *Physical Review A* 44 (1991) 1182–1202.
- [69] D. Salac, M. Miksis, Reynolds number effects on lipid vesicles, *Journal of Fluid Mechanics* 711 (2012) 122–146.
- [70] P. Song, D. Hu, P. Zhang, Numerical simulation of fluid membranes in two-dimensional space, *Communications in Computational Physics* 3 (2008) 794–821.
- [71] Z. Tan, D. V. Le, Z. Li, K. M. Lim, B. C. Khoo, An immersed interface method for solving incompressible viscous flows with piecewise constant viscosity across a moving elastic membrane, *Journal of Computational Physics* 227 (2008) 9955–9983.
- [72] C. Tu, C. S. Peskin, Stability and instability in the computation of flows with moving immersed boundaries: a comparison of three methods, *SIAM Journal on Scientific Computing* 13 (1992) 1361–1378.
- [73] H. Lamb, *Hydrodynamics*, 6th ed., Cambridge University Press, 1932.
- [74] A. S. Nesbitt, E. Westein, F. J. Tovar-Lopez, E. Tolouei, A. Mitchell, J. Fu, J. Carberry, A. Fouras, S. P. Jackson, A shear gradient-dependent platelet aggregation mechanism drives thrombus formation, *Nature Medicine* 15 (2009) 665–675.
- [75] D. Isèbe, P. Nérin, Numerical simulation of particle dynamics in an orifice-electrode system. Application to counting and sizing by impedance measurement, *International Journal for Numerical Methods in Biomedical Engineering* 29 (2013) 462–475.
- [76] B. Kaoui, G. H. Ristow, I. Cantat, C. Misbah, W. Zimmermann, Lateral migration of a two-dimensional vesicle in unbounded Poiseuille flow, *Physical Review E* 77 (2008) 021903.