Commun. Comput. Phys.
doi: 10.4208/cicp.200110.040511a

Vol. **11**, No. 3, pp. 925-950
March 2012

An Immersed Interface Method for the Simulation of Inextensible Interfaces in Viscous Fluids

Zhijun Tan[1,*], D. V. Le[2], K. M. Lim[3,4] and B. C. Khoo[3,4]

[1] *Guangdong Province Key Laboratory of Computational Science & School of Mathematics and Computational Science, Sun Yat-sen University, Guangzhou 510275, China.*
[2] *Institute of High Performance Computing, 1 Fusionopolis Way, #16-16 Connexis, Singapore 138632, Singapore.*
[3] *Singapore-MIT Alliance, 4 Engineering Drive 3, National University of Singapore, Singapore 117576, Singapore.*
[4] *Department of Mechanical Engineering, National University of Singapore, 10 Kent Ridge Crescent, Singapore 119260, Singapore.*

Received 20 January 2010; Accepted (in revised version) 4 May 2011

Communicated by Jaw-Yen Yang

Available online 28 October 2011

Abstract. In this paper, an immersed interface method is presented to simulate the dynamics of inextensible interfaces in an incompressible flow. The tension is introduced as an augmented variable to satisfy the constraint of interface inextensibility, and the resulting augmented system is solved by the GMRES method. In this work, the arclength of the interface is locally and globally conserved as the enclosed region undergoes deformation. The forces at the interface are calculated from the configuration of the interface and the computed augmented variable, and then applied to the fluid through the related jump conditions. The governing equations are discretized on a MAC grid via a second-order finite difference scheme which incorporates jump contributions and solved by the conjugate gradient Uzawa-type method. The proposed method is applied to several examples including the deformation of a liquid capsule with inextensible interfaces in a shear flow. Numerical results reveal that both the area enclosed by interface and arclength of interface are conserved well simultaneously. These provide further evidence on the capability of the present method to simulate incompressible flows involving inextensible interfaces.

AMS subject classifications: 65N06, 35R05, 65M12

Key words: Inextensible interface, Stokes flows, singular force, immersed interface method, CG-Uzawa method, front tracking.

*Corresponding author. *Email addresses:* tzhij@mail.sysu.edu.cn (Z.-J. Tan), ledv@ihpc.a-star.edu.sg (D. V. Le), mpelimkm@nus.edu.sg (K. M. Lim), mpekbc@nus.edu.sg (B. C. Khoo)

1 Introduction

The membrane of biological cells consisting of lipid bilayers has much attention due to the occurrence in many biological phenomena [28] and used widely as model for the red blood cells [24] and drug-carrying capsules [30]. Most biological membranes can deform but resist area dilation and are often modelled as inextensible interfaces with the position-dependent tension playing the role of surface pressure [41]. To account for the interface inextensibility or incompressibility, the tension is an unknown quantity which is to be computed as part of the solution so as to satisfy the condition of inextensibility. In another word, this is to ensure that the arclength of an arbitrary element of the membrane is conserved during the motion. The viscous flows outside and inside the membrane can also be treated as comprising similar or different incompressible fluids.

Peskin's immersed boundary method (IBM) [22] has been applied widely for simulating such biological flows with moving interfaces. The method was originally developed to study the fluid dynamics of blood flow in a human heart [21], and has further been developed for a wide variety of applications; in particular for the biological problems where complex geometries and immersed elastic interfaces are present, such as the deformation of red blood cells in a shear flow [7], swimming of organisms [9], platelet aggregation [10, 11, 37], and cochlear dynamics. Other applications can be found in [2], biofilm processes [6], wood pulp fiber dynamics [25], and with a more extensive list given in [22]. In the IBM, the force densities are spread to the Cartesian grid points by a discrete representation of the delta function. The fluid equations with the forcing terms are then solved for the pressure and the velocity at the mentioned Cartesian grid points. The resulting velocities are then interpolated back to the control points using the same set of discrete delta functions. Since the immersed boundary method uses the discrete delta function approach, it smears out sharp interfaces and it is of first-order accuracy in space.

In order to capture the jumps in the solution across the interface, the immersed interface method (IIM) incorporates the known jumps into the finite difference scheme near the interface. As such, the IIM avoids smearing out the sharp interfaces and maintains a second-order accuracy. The IIM was originally proposed by LeVeque and Li [15] for solving elliptic equations, and later extended to Stokes flow with elastic boundaries or surface tension [14]. The method was further developed for the Navier-Stokes equations in [12, 13, 17, 19, 39]. The IIM was also used in [4, 18, 26] for solving the two-dimensional streamfunction-vorticity equations on irregular domains. Tan et al. [36] developed an IIM for the Stokes equations on irregular domain. In [35], Tan et al. developed an IIM for the Navier–Stokes equations with discontinuous viscosity across the interface. Xu and Wang [40] extended the IIM approach to the 3D Navier-Stokes equation for simulating the fluid-solid interaction. The interested readers are referred to the recently published book by Li and Ito [16] and the references therein for more applications of the IIM.

In [41], Zhou and Pozrikidis studied the deformation of inextensible interfaces based on the boundary element formulation. A boundary integral method developed in [33] for simulating the dynamics of inextensible vesicles suspended in a viscous fluid in 2D is

further extended to the 3D axisymmetric vesicles [34] recently. To our knowledge, there is no published work in the literature on IIM for solving the dynamics of inextensible interface in an incompressible fluid. As such, it would be interesting to carry out an IIM for solving the Stokes flows involving inextensible interfaces as in this paper. The present IIM significantly differs from those in [33,41] and requires no Green function and is very flexible in terms of (possible) different boundary conditions compared with theirs. Also the boundary integral method cannot be directly applied to solve for the Navier–Stokes equations. It should be reiterated in this work the difference or the special feature is the area conservation enclosed by the interface and the inextensibility of the interface are maintained simultaneously in contrast to the published works on IIM [12–14,17,35,39].

In the present study, the proposed method combines the IIM with a front tracking representation of the interface on a uniform Cartesian grid. In order to employ the IIM for solving the inextensible interface problem, the interface tension is introduced as the augmented variable to maintain the condition of interface inextensibility. The resulting augmented system is then solved by the GMRES method such that a differential element of the interface deforms while maintaining the original area enclosed by the interface and arclength of the interface. Based on this augmented variable, the forces associated with the inextensible interfaces exerting on the fluid are computed from the configuration/shape of the interfaces. These forces are interpolated using cubic splines and related to the jump in pressure and the jumps in the derivatives of velocity and pressure, and then applied to the fluid through the jump conditions. The position of the deformed interfaces is updated implicitly using the BFGS method within each time step. The Stokes equations are discretized on a staggered Cartesian grid by a second order finite difference method and solved by the conjugate gradient Uzawa-type method. The jumps in the solution and its derivatives are incorporated into the finite difference discretization to obtain a sharp interface resolution. Fast solvers from the FISHPACK software library [1] are then used to solve the resulting discrete systems of the Poisson equations. The capability of the proposed IIM to simulate the deformation of elastic capsules with inextensible interface is demonstrated by some numerical examples. The numerical results indicate that the present method can achieve second order accuracy for the velocity and close to second-order accuracy for the pressure.

The remaining part of the paper is organized as follows. In Section 2, the governing equations and problem description are presented. The jump conditions across the inextensible interface are presented in Section 3. The numerical algorithm and implementations are presented in Section 4 and 5, respectively. In Section 6, two extensive numerical experiments are included. Finally, concluding remarks are made in Section 7.

2 Governing equations

This paper concerns the 2D viscous incompressible Stokes flows with inextensible interface. In a two dimensional bounded domain Ω, which contains a closed interface Γ,

the incompressible steady Stokes equations formulated in the primitive velocity-pressure variables is considered and written as

$$\nabla p = \mu \Delta \mathbf{u} + \mathbf{F}(\mathbf{x},t), \quad \mathbf{x} \in \Omega, \tag{2.1}$$

$$\nabla \cdot \mathbf{u} = 0, \quad \mathbf{x} \in \Omega, \tag{2.2}$$

with the Dirichlet boundary condition

$$\mathbf{u}|_{\partial\Omega} = \mathbf{u}_b, \tag{2.3}$$

where $\mathbf{u} = (u,v)^T$ is the fluid velocity, p is the pressure, μ is the fluid viscosity (assumed to be same across the interface) and $\mathbf{x} = (x,y)$ is the Cartesian coordinate variables. The effect of the interface Γ results in a singular force $\mathbf{F}$ which is not zero only at the interface and can be expressed as

$$\mathbf{F}(\mathbf{x}) = \int_\Gamma \mathbf{f}(s,t)\delta(\mathbf{x} - \mathbf{X}(s,t))ds. \tag{2.4}$$

Here $\mathbf{X}(s,t)$ denotes the location of the interface Γ at time t, parameterized by the arc-length s. $\mathbf{f} = (f_1,f_2)^T$ is the force density, and $\delta(\cdot)$ denotes the two-dimensional Dirac delta function defined in the distribution sense. Eq. (2.2) together with the Dirichlet boundary condition Eq. (2.3) leads to the compatibility condition that $\mathbf{u}_b$ must satisfy:

$$\int_{\partial\Omega} \mathbf{u}_b \cdot \mathbf{n}_b dS = 0, \tag{2.5}$$

where $\mathbf{n}_b$ is the outer unit normal to $\partial\Omega$. The motion of the interface satisfies

$$\frac{\partial \mathbf{X}(s,t)}{\partial t} = \mathbf{u}(\mathbf{X}(s,t),t) = \int_\Omega \mathbf{u}(\mathbf{x},t)\delta(\mathbf{x} - \mathbf{X}(s,t))d\mathbf{x}. \tag{2.6}$$

The tangential vector on the interface Γ is given by $\boldsymbol{\tau}(s,t)$ with the form of

$$\boldsymbol{\tau}(s,t) = \frac{\partial \mathbf{X}}{\partial s} \Big/ \Big|\frac{\partial \mathbf{X}}{\partial s}\Big|. \tag{2.7}$$

In the present study, a fluid problem involving a closed inextensible interface is considered, and the inextensibility constraint for an evolving interface in 2D is expressed by the equation

$$\nabla_s \cdot \mathbf{u}|_\Gamma = \boldsymbol{\tau} \cdot \frac{\partial \mathbf{u}}{\partial \boldsymbol{\tau}} = 0. \tag{2.8}$$

This is due to the fact that the material derivative of an arclength element (i.e. the local stretching factor) for inextensible interface satisfies,

$$0 = \frac{d}{dt}|\mathbf{X}_s| = \nabla_s \cdot \mathbf{u}|\mathbf{X}_s|. \tag{2.9}$$

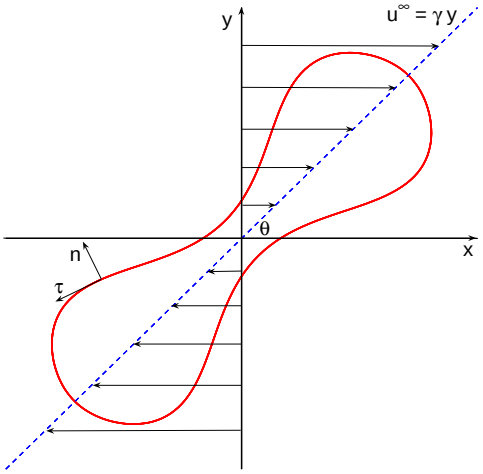

Figure 1: Schematic illustration of a 2D capsule in shear flow.

The force strength $\mathbf{f}$ exerted on the fluid by the interface has the form of

$$\mathbf{f}(s,t)=\frac{\partial}{\partial s}\Big(T(s,t)\boldsymbol{\tau}(s,t)\Big), \tag{2.10}$$

with the surface tension $T(s,t)$ as in [41]. Thus, the force density can be computed directly from the location $\mathbf{X}$ of the interface Γ. An equivalent form of Eq. (2.10) can be denoted as

$$\mathbf{f}(s,t)=(\partial T/\partial s)\boldsymbol{\tau}(s,t)+T\kappa\mathbf{n}, \tag{2.11}$$

where κ is the curvature defined by $\partial\boldsymbol{\tau}/\partial s=\kappa\mathbf{n}$. It is noted that the other forces (e.g., the bending force $\mathbf{f}_b=-c_b(\kappa_{ss}+\frac{1}{2}\kappa^3)\mathbf{n}$ in [33], where c_b is a bending coefficient) can be easily incorporated into the right hand side of Eq. (2.11). In the final part of the paper, the effect of the bending is also tested by some cases. Note that, unlike the previous work on IIM for moving interface in which the tension T is a function of the location of the interface, the tension quantity is unknown here. One of the challenges for such a problem is that the pressure and the tension are coupled implicitly by the fluid's incompressible condition (2.2), the interface's inextensible condition (2.8) and no-slipping condition for motion of interface (2.6). The introduction of the above scalar tension provides us with a degree of freedom that allows the satisfaction of the inextensibility constraint (2.8) at every point on the interface. In this work, the deformation of a 2D capsule subject to a incident shear flow along the x-axis, $\mathbf{u}^\infty=(\gamma y,0)$, is considered, where γ is the shear rate. The readers are referred to Fig. 1 for a schematic illustration of the problem, where θ is orientation angle, and $\mathbf{n}=(n_1,n_2)$ and $\boldsymbol{\tau}=(\tau_1,\tau_2)$ are the unit outward normal and tangential vector to the interface, respectively.

3 Jump conditions across the interface

The jump of an arbitrary function $w(\mathbf{X})$ across the interface at $\mathbf{X}$ is denoted by

$$[w]=\lim_{\epsilon\to 0^+} w(\mathbf{X}+\epsilon\mathbf{n})-\lim_{\epsilon\to 0^+} w(\mathbf{X}-\epsilon\mathbf{n}). \tag{3.1}$$

Introducing the tension at the interface as the augmented variable, i.e., $q(s)=T(X(s,t),Y(s,t))$, and denoting (ξ,η) the local coordinates associated with the directions of $\mathbf{n}$ and $\boldsymbol{\tau}$, respectively, the jump conditions for the velocity and pressure across the interface (see [12,17] for details) are given/obtained as follows:

$$[\mathbf{u}]=\mathbf{0},\quad [\mathbf{u}_\eta]=\mathbf{0},\quad [\mathbf{u}_\xi]=-\frac{1}{\mu}\hat{f}_2\boldsymbol{\tau}, \tag{3.2}$$

$$[\mathbf{u}_{\eta\eta}]=\frac{1}{\mu}\kappa\hat{f}_2\boldsymbol{\tau},\quad [\mathbf{u}_{\xi\eta}]=-\frac{1}{\mu}\frac{\partial\hat{f}_2}{\partial\eta}\boldsymbol{\tau}-\frac{1}{\mu}\kappa\hat{f}_2\mathbf{n}, \tag{3.3}$$

$$[\mathbf{u}_{\xi\xi}]=-[\mathbf{u}_{\eta\eta}]+\frac{1}{\mu}[p_\xi]\mathbf{n}+\frac{1}{\mu}[p_\eta]\boldsymbol{\tau}, \tag{3.4}$$

$$[p]=\hat{f}_1,\quad [p_\xi]=\frac{\partial\hat{f}_2}{\partial\eta},\quad [p_\eta]=\frac{\partial\hat{f}_1}{\partial\eta}, \tag{3.5}$$

$$[p_{\eta\eta}]=\frac{\partial^2\hat{f}_1}{\partial\eta^2}-\kappa[p_\xi],\quad [p_{\xi\eta}]=\frac{\partial^2\hat{f}_2}{\partial\eta^2}+\kappa[p_\eta], \tag{3.6}$$

$$[p_{\xi\xi}]=-[p_{\eta\eta}]. \tag{3.7}$$

Here, $\hat{f}_1$ and $\hat{f}_2$ are the components of the force density in the normal and tangential directions of the interface such that $\hat{\mathbf{f}}=(\hat{f}_1,\hat{f}_2)=(\partial T/\partial s,T\kappa)$, and κ is the signed curvature of the interface. It is noted from expressions (3.2)-(3.7) that the values of the jumps of the first and second order derivatives of velocity and pressure can be obtained by a simple coordinate transformation:

$$[w_x]=[w_\xi]n_1+[w_\eta]\tau_1,\qquad [w_y]=[q_\xi]n_2+[w_\eta]\tau_2, \tag{3.8}$$

$$[w_{xx}]=[w_{\xi\xi}]n_1^2+2[w_{\xi\eta}]n_1\tau_1+[w_{\eta\eta}]\tau_1^2, \tag{3.9}$$

$$[w_{yy}]=[w_{\xi\xi}]n_2^2+2[w_{\xi\eta}]n_2\tau_2+[w_{\eta\eta}]\tau_2^2,\qquad w=\mathbf{u},p. \tag{3.10}$$

4 Numerical algorithm

The numerical algorithm to be employed is based on the conjugate gradient Uzawa-type method for the discretization of the Stokes equations with special treatment at the grid points near the interface. The spatial discretization is carried out on a standard marker-and-cell (MAC) staggered grid similar to that found in Tau [32] with mesh size $h=\Delta x=\Delta y$. With the MAC mesh, the pressure field is defined at the cell center (i,j), where $i\in$

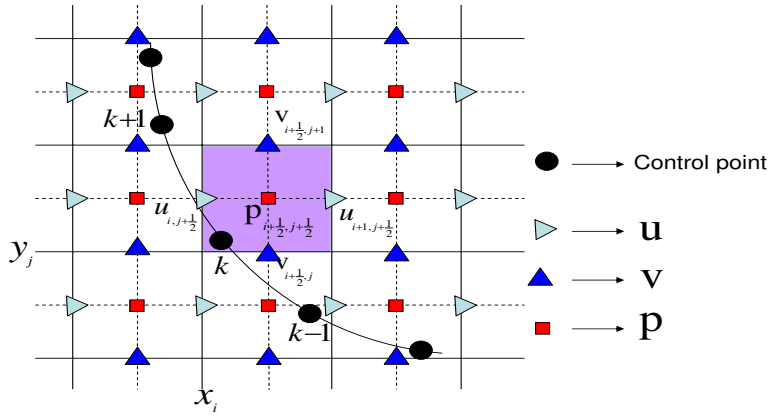

Figure 2: A diagram of the interface cutting through a staggered grid with a uniform mesh size h, where the velocity component u is at the left-right face of the cell and v is at the top-bottom face, and the pressure is at the cell center.

$\{1,2,\cdots,N_x\}$ and $j\in\{1,2,\cdots,N_y\}$. The velocity fields u and v are defined at the vertical and horizontal edges of a cell, respectively. The pressure and the velocity components u and v are arranged as depicted in Fig. 2. An advantage of such a staggered grid is that there is no need for pressure boundary conditions when dealing with the derivative of pressure since the pressure nodes are at the cell center.

4.1 Stokes solver involving correction terms

The discretization of Eqs. (2.1)-(2.3) by the second order MAC finite difference scheme leads to the following linear system

$$-\mu\Big(\frac{u_{i+1,j+\frac{1}{2}}-2u_{i,j+\frac{1}{2}}+u_{i-1,j+\frac{1}{2}}}{h^2}+\frac{u_{i,j+\frac{3}{2}}-2u_{i,j+\frac{1}{2}}+u_{i,j-\frac{1}{2}}}{h^2}\Big)-\mu C\{\Delta u\}_{i,j}$$
$$+\frac{p_{i+\frac{1}{2},j+\frac{1}{2}}-p_{i-\frac{1}{2},j+\frac{1}{2}}}{h}+C\{p_x\}_{i,j}=g^1_{i,j}, \tag{4.1}$$
$$-\mu\Big(\frac{v_{i+\frac{3}{2},j}-2v_{i+\frac{1}{2},j}+v_{i-\frac{1}{2},j}}{h^2}+\frac{v_{i+\frac{1}{2},j+1}-2v_{i+\frac{1}{2},j}+v_{i+\frac{1}{2},j-1}}{h^2}\Big)-\mu C\{\Delta v\}_{i,j}$$
$$+\frac{p_{i+\frac{1}{2},j+\frac{1}{2}}-p_{i+\frac{1}{2},j-\frac{1}{2}}}{h}+C\{p_y\}_{i,j}=g^2_{i,j}, \tag{4.2}$$
$$\frac{u^{i+1,j+\frac{1}{2}}-u^{i,j+\frac{1}{2}}}{h}+\frac{v^{i+\frac{1}{2},j+1}-v^{i+\frac{1}{2},j}}{h}+C\{\nabla\cdot\mathbf{u}\}_{i,j}=0. \tag{4.3}$$

Note that the discretization of the Stokes equations at the grid points near the interface has been modified to account for the jump conditions across the interface due to the presence of singular forces at the interface. In Eqs. (4.1)-(4.3), $C\{\Delta u\}_{i,j}$, $C\{\Delta v\}_{i,j}$, $C\{p_x\}_{i,j}$, $C\{p_y\}_{i,j}$, and $C\{\nabla\cdot\mathbf{u}\}_{i,j}$ are just the corresponding spatial operator correction terms, which are added to the finite difference equations and only non-zero at those points near

the interface, to improve the accuracy of the local finite difference approximations. These corrections will be evaluated later. In order to satisfy the discrete compatibility condition corresponding to (2.5) to thereby ensure the solvability of system Eqs. (4.1)-(4.3), a solvable perturbed system in a way similar to that found in [14] via perturbing $C\{\nabla\cdot\mathbf{u}\}_{i,j}$ to $\hat{C}\{\nabla\cdot\mathbf{u}\}_{i,j}$ in Eq. (4.3) is employed. Here, $\hat{C}\{\nabla\cdot\mathbf{u}\}_{i,j}=C\{\nabla\cdot\mathbf{u}\}_{i,j}-\bar{C}\{\nabla\cdot\mathbf{u}\}_{i,j}$, where $\bar{C}\{\nabla\cdot\mathbf{u}\}_{i,j}$ is the mean value of the correction term $C\{\nabla\cdot\mathbf{u}\}_{i,j}$. The readers are referred to [14] for details. Let Δ_h, G^{MAC}, and D^{MAC} be the standard central difference operator, the MAC gradient, and the divergence operators, respectively, then system (4.1)-(4.3) can be written as

$$-\mu\Delta_h\mathbf{u}+G^{\mathrm{MAC}}p=\mathbf{g}+\boldsymbol{C}_1, \tag{4.4}$$

$$D^{\mathrm{MAC}}\mathbf{u}=C_2-\bar{C}_2, \tag{4.5}$$

where the coefficients $\boldsymbol{C}_1$ and C_2 are the spatial correction terms whose expressions will be given in the next subsection, and $\bar{C}_2$ is the perturbing term. Let $\mathbf{G}_1=\mathbf{g}(\mathbf{x})+\boldsymbol{C}_1$ and $G_2=C_2-\bar{C}_2$, then (4.4)-(4.5) can be written in the matrix-vector form as

$$\begin{pmatrix} -\mu\Delta_h & G^{\mathrm{MAC}} \\ D^{\mathrm{MAC}} & 0 \end{pmatrix}\begin{pmatrix} \mathbf{u} \\ p \end{pmatrix}=\begin{pmatrix} \mathbf{G}_1 \\ G_2 \end{pmatrix}. \tag{4.6}$$

There are some fast solvers for the solution of (4.6), such as the PCG method [8, 23], the PMINRES method [8, 23], the FFT-based method [5], and the multigrid method [8, 20, 23]. In this work, the fast solvers from FISHPACK [1] are utilized to incorporate the CG-Uzawa method. The Uzawa procedure for problems with immersed interfaces is analogous to the fast iterative method presented in [29, 32] and it consists of two steps:

Step 1. Solve $D^{\mathrm{MAC}}\Delta_h^{-1}G^{\mathrm{MAC}}p=\mu G_2+D^{\mathrm{MAC}}\Delta_h^{-1}\mathbf{G}_1$ for the pressure p,

Step 2. Solve $\mu\Delta_h\mathbf{u}=\mathbf{G}_1-G^{\mathrm{MAC}}p$ for the velocity $\mathbf{u}$.

Here, $D^{\mathrm{MAC}}\Delta_h^{-1}G^{\mathrm{MAC}}$ is the Schur complement of system (4.6). In Step 1, the system is solved by the conjugate gradient method (CG) in this work. In the CG method, each matrix-vector product with $D^{\mathrm{MAC}}\Delta_h^{-1}G^{\mathrm{MAC}}$ requires the inverse of Δ_h which corresponds to solving a Poisson equation. Several fast methods can be applied, such as the ICCG method, the FFT method and multigrid method. In the present work, the fast solvers from FISHPACK [1] are used. Once the pressure is obtained, the velocity field $\mathbf{u}$ can be solved by the fast solvers from FISHPACK [1] via Step 2.

4.2 Calculation of correction terms

The correction terms $\boldsymbol{C}_1$ and C_2 are evaluated as follows:

$$\boldsymbol{C}_1=\mu\Big(C\{\Delta\mathbf{u}\}\Big)-C\{\nabla p\}, \tag{4.7a}$$

$$C_2=-C\{\nabla\cdot\mathbf{u}\}. \tag{4.7b}$$

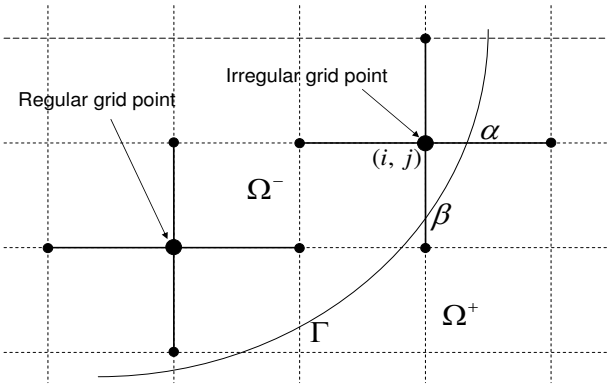

Figure 3: Interface and mesh geometry near the irregular grid point (i,j).

To evaluate the correction term $C\{\Delta\mathbf{u}\}$ of (4.7a) at an irregular point (i,j) as depicted in Fig. 3, the jump conditions $[\mathbf{u}_x]$ and $[\mathbf{u}_{xx}]$ at the intersection point α of the interface with a grid line, and $[\mathbf{u}_y]$ and $[\mathbf{u}_{yy}]$ at β of the interface with a grid line, need to be computed. The correction term $C\{\Delta\mathbf{u}\}$ is calculated as follows:

$$C\{\Delta\mathbf{u}\}_{i,j}=-\frac{[\mathbf{u}]+h^+[\mathbf{u}_x]_\alpha+\frac{(h^+)^2}{2}[\mathbf{u}_{xx}]_\alpha}{h^2}-\frac{[\mathbf{u}]+k^-[\mathbf{u}_y]_\beta+\frac{(k^-)^2}{2}[\mathbf{u}_{yy}]_\beta}{h^2},$$

where $h^+=x_{i+1}-x_\alpha$, $k^-=y_{j-1}-y_\beta$, and x_α and y_β are the x-coordinate of the intersection point α and the y-coordinate of the intersection point β as shown in Fig. 3, respectively. $\Delta\mathbf{u}$ is approximated at the irregular point (i,j) as

$$\Delta\mathbf{u}(i,j)=\Delta_h\mathbf{u}_{i,j}+C\{\Delta\mathbf{u}\}_{i,j}+\mathcal{O}(h).$$

Similarly, the other correction terms in (4.7a)-(4.7b) can be computed as follows

$$C\{\nabla\cdot\mathbf{u}\}_{i,j}=-\frac{[u]+h^+[u_x]_\alpha+\frac{(h^+)^2}{2}[u_{xx}]_\alpha}{h}+\frac{[v]+k^-[v_y]_\beta+\frac{(k^-)^2}{2}[v_{yy}]_\beta}{h},$$
$$C\{\nabla p\}_{i,j}=\left(-\frac{[p]+h^+[p_x]_\alpha+\frac{(h^+)^2}{2}[p_{xx}]_\alpha}{h},\frac{[p]+k^-[p_y]_\beta+\frac{(k^-)^2}{2}[p_{yy}]_\beta}{h}\right).$$

4.3 Determination of the tension at control points

Assuming that the tension q at the interface is known, the velocity field $\mathbf{u}$ at all the grid points can be computed via the CG-Uzawa method as discussed in Section 4.1. The velocity at the control points, $\mathbf{U}_k$, can be interpolated from the velocity $\mathbf{u}$ at the grid points as shown in [12], which can be written as

$$\mathbf{U}_k=\mathbf{U}(\mathbf{X}_k)=\mathcal{B}(\mathbf{u}), \tag{4.8}$$

where $\mathcal{B}$ is the interpolation operator which includes the appropriate correction terms [12]. With this velocity at the control points, then the surface divergence at the control points can be obtained from the formula $\nabla_s \cdot \mathbf{U}_k = \boldsymbol{\tau} \cdot \frac{\partial \mathbf{U_k}}{\partial \boldsymbol{\tau}}$. Since the relationships for the tension, the singular forces and the jumps in the solution or its derivatives are discretized linearly and all the equations solved are linear, the surface divergence of the velocity at the interface can be written as,

$$\nabla_s \cdot \mathbf{U}_k = \nabla_s \cdot \mathbf{U}_k^0 + \mathbf{A}\mathbf{q}, \tag{4.9}$$

where $\mathbf{U}_k^0$ corresponds to the velocity at the control points obtained by solving Eqs. (2.1) and (2.2) with $q = 0$ (i.e. the tension $T = 0$) at the control points. $\mathbf{A}$ is a $N_b \times N_b$ matrix, where N_b is the number of control points. The vector $\mathbf{A}\mathbf{q}$ is the surface divergence of the velocity at the control points obtained by solving the following equations:

$$\nabla_h p_q = \mu \Delta_h \mathbf{u}_q + \check{\boldsymbol{C}}_1, \tag{4.10}$$

$$\nabla_h \cdot \mathbf{u}_q = \check{C}_2, \quad \mathbf{u}_q|_{\partial\Omega} = 0, \tag{4.11}$$

$$\mathbf{A}\mathbf{q} = \nabla_s \cdot \mathcal{B}(\mathbf{u}_q), \tag{4.12}$$

with q being the tension at the interface. Here, $\check{\boldsymbol{C}}_1$ and $\check{C}_2$ are the correction terms which take into account the effect of the tension q at the interface. From Eq. (4.9), with the inextensibility constraint at the interface (2.8), the tension q at the interface is determined by solving

$$\mathbf{A}\mathbf{q} = \nabla_s \cdot \mathbf{U}_k - \nabla_s \cdot \mathbf{U}_k^0 = -\nabla_s \cdot \mathbf{U}_k^0. \tag{4.13}$$

Note that the matrix $\mathbf{A}$ depends on the location of the interface. In the present work, the interface is moving, and the system of Eq. (4.13) can be solved by the GMRES method [27], which only requires the matrix vector multiplication.

4.4 Advancing the interface via the BFGS method

The location of the interface $\mathbf{X}$ is updated based on the surrounding fluid velocity. To overcome the strict limit of very small time steps in an explicit scheme and increase the stability of the current method, the updated location of the deformable interface is advanced implicitly in time in the following way:

$$\mathbf{X}^{n+1} = \mathbf{X}^n + \frac{1}{2}\Delta t \Big(\mathbf{u}^n(\mathbf{X}^n) + \mathbf{u}^{n+1}(\mathbf{X}^{n+1}) \Big). \tag{4.14}$$

Calculating the new positions of the control points $\mathbf{X}^{n+1}$ is equivalent to seeking the solution of the nonlinear equation

$$Q(\mathbf{X}^{n+1}) = 0, \tag{4.15}$$

where

$$Q(\mathbf{X}) = \mathbf{X} - \mathbf{X}^n - \frac{1}{2}\Delta t\Big(\mathbf{u}^n(\mathbf{X}^n) + \mathbf{u}^{n+1}(\mathbf{X})\Big).$$

The BFGS method [31] is employed to solve the nonlinear system of Eq. (4.14) iteratively to calculate the location of the moving interface. For more details on the immersed interface method for moving interfaces, see [12–14]. In each iteration of the BFGS method, the system of Eq. (4.13) for the augmented variable need to be solved to satisfy the condition of the interface inextensibility.

5 Numerical implementation

In this section, the implementation of the proposed algorithm is described. Given the location of the control points $\mathbf{X^n}$, the velocity $\mathbf{u}^n$ and the pressure p^n, the algorithm for computing the velocity $\mathbf{u}^{n+1}$, pressure p^{n+1} and the location of the control points $\mathbf{X}^{n+1}$ can be described as follows:

IIM Algorithm for flow with inextensible interface:

Step 1: Set $k:=0$, make an initial guess for $\mathbf{X}^{n+1}$, i.e. $\mathbf{X}^{(0)}$ as

$$\mathbf{X}^{(0)} = 2\mathbf{X}^n - \mathbf{X}^{n-1}.$$

Step 2:

- Compute the augmented variable $\mathbf{q}$ at the interface to satisfy the condition of interface inextensibility. That is, calculate the right hand side vector of (4.13). Then solve for the small system of Eq. (4.13) using GMRES method to obtain the augmented variable $\mathbf{q}$ along the interface.
- Compute the force strength $\mathbf{f}$ at the inextensible interface using expression (2.10).
- Employ the CG-Uzawa method as described in Section 4.1 to obtain the velocity field $\mathbf{u}^{n+1}$ and pressure field p^{n+1}. This step involves computing the appropriate correction terms for the spatial derivatives as described in Section 4.2.
- Compute the velocity $\mathbf{u}^{n+1}(\mathbf{X}^{(k)})$ at control points $\mathbf{X}^{(k)}$, which are interpolated from the velocity $\mathbf{u}^{n+1}$ at the surrounding grid points.

Step 3:

- Evaluate $Q(\mathbf{X}^{(k)})$.
- If $\|Q^{(k)}\| < \epsilon$ then $\mathbf{X}^{n+1} = \mathbf{X}^{(k)}$ and stop the iteration. Otherwise, update $\mathbf{X}^{(k+1)}$ and the inverse Jacobian matrix B_{k+1}^{n+1} [31]. Set $k = k+1$ and go to **Step** 2.

6 Numerical experiments

In this section, two numerical experiments are carried out to demonstrate the capabilities of the proposed method.

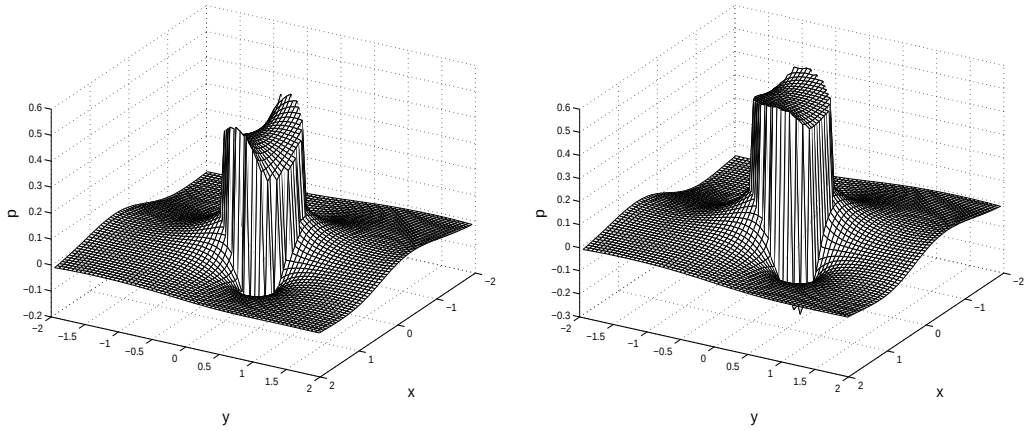

Figure 4: For Example 6.1. Pressure profiles at $t=0.5$ (left) and $t=2$ (right).

Example 6.1. An initially elliptical capsule

In this example, the deformation of an initially elliptical capsule is simulated in shear flow, where the shear rate is $\gamma=1$. The capsule initially has the semi-major axis $a=0.75$ and semi-minor axis $b=0.5$ and is located at the center of the square domain $[-4,4]\times[-2,2]$.

In the simulations, a 128×64 grid system is employed, and 48 control points are used to represent the interface. The fluid viscosity $\mu=0.1$ and the time step $\Delta t=10^{-3}$ are used. The initial inclination angle of the capsule to the flow is $\theta_0=0$. The pressure profiles at $t=0.5$ and $t=2$ are presented in Fig. 4 (left) and Fig. 4 (right), respectively. As expected, it can be observed from these figures that the pressure is discontinuous across the interface, however the jump in pressure is very sharply captured by the present method. Fig. 5 shows the shapes of deformed capsule at different times. It can be seen from this figure that the capsule has only small deformation and reaches a steady shape.

The plot of streamlines at the steady state is presented in Fig. 6 (left), where the bold closed solid line represents the configuration of the capsule. Single eddies formed inside the capsule and recirculating regions are observed in this figure. The capsule rotates around the liquid inside and the fluid flow inside the capsule moves in the clockwise direction. In Fig. 6 (left), a closed streamline is aligned with the capsule interface, which indicates a steady-state shape of capsule is achieved at this moment. The corresponding pressure profile at the steady state is presented in Fig. 6 (right).

The distribution of the interface tension T and tank-treading velocity (i.e., the tangential component of the capsule velocity, $\mathbf{u}\cdot\boldsymbol{\tau}$) versus arclength at steady state are plotted in Fig. 7 (left) and Fig. 7 (right), respectively. It is observed from Fig. 7 (right) that tank-treading velocity is constant along the interface of the capsule as required by the condition of interface inextensibility, which shows that the capsule engages in a tank-treading motion around the capsule with uniform tangential velocity. But the tension along the in-

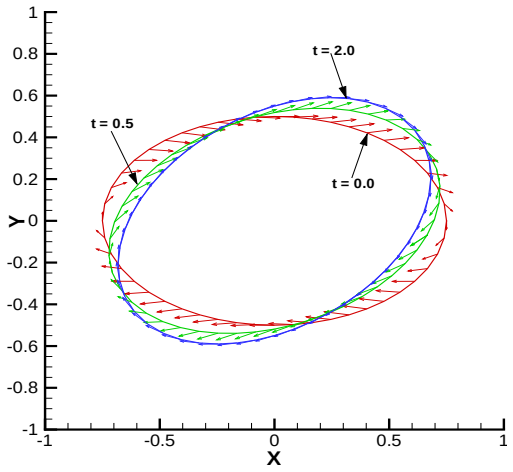

Figure 5: For Example 6.1. Shapes of deformed capsules at different times. The velocities at the control points of inextensible interfaces are shown as the arrows.

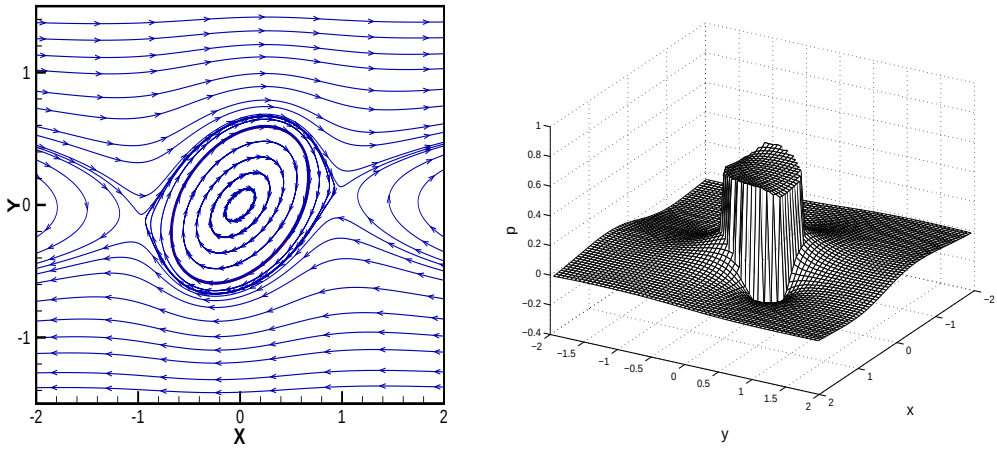

Figure 6: For Example 6.1. Streamline pattern and pressure profile at steady state.

terface show significant variations as seen from Fig. 7 (left), where the distribution of the tension shows a nearly sinusoidal variation, and the points of maximum and minimum tension are located near the points of minimum and maximum curvature.

Since the fluid is incompressible and the interface is inextensible, the length of the capsule and the enclosed area should remain constant throughout the simulation. Let A_0 and A_t be the enclosed area by the capsule at the initial time and at time t, respectively. The maximum error in the enclosed area is defined as $\max_t|A_t-A_0|$, and the area loss rate is defined as $\max_t|A_t-A_0|/A_0$. The similar definition is for arclength. In Fig. 8 (left), the plot of the area conservation error versus time up to $t=3$ is presented for $\theta_0=0$. From the figure, the maximum error in the area is less than $1.5e-004$ and it indicates only a very small area loss rate which is less than 0.01273%. The plot of arclength conservation

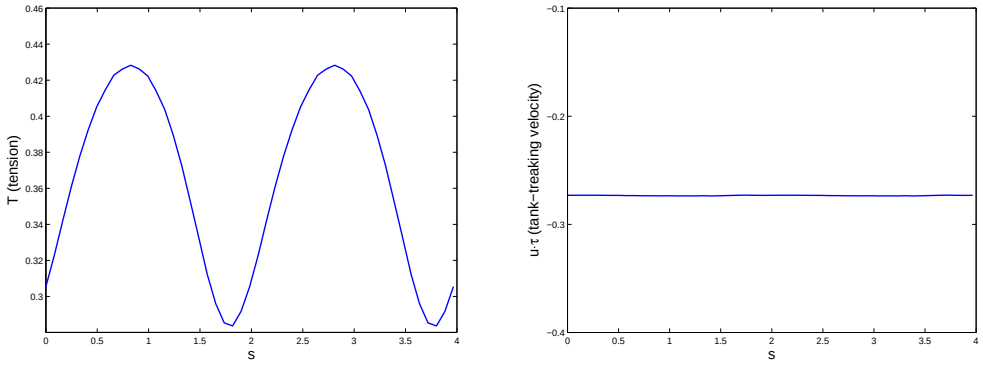

Figure 7: For Example 6.1. The interface tension (left) and tank-treading velocity (right) versus arc length at steady state.

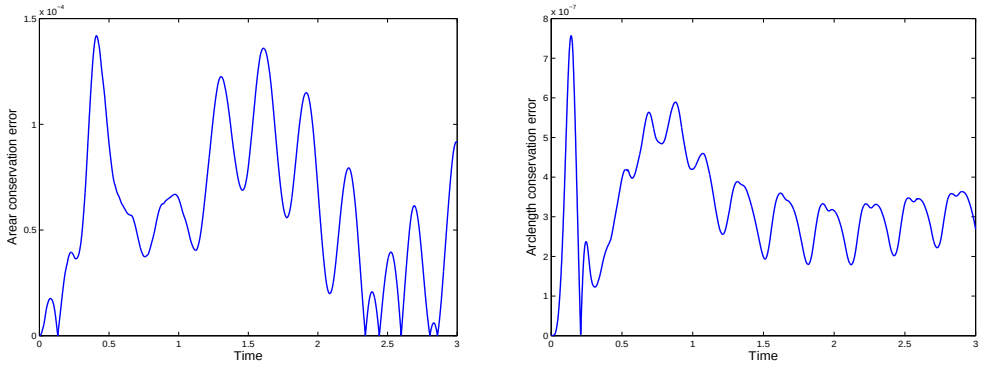

Figure 8: For Example 6.1. The area conservation error (left) and arclength conservation error (right) versus time.

error versus time is presented in Fig. 8 (right), and the error in the capsule arclength is preserved within $8e-07$. These figures show that the area enclosed by the capsule and the arclength of capsule are both conserved very well, which results from the requirements that the fluid enclosed in the capsule is incompressible and the interface of the capsule is locally inextensible.

The deformation of an initially elliptical capsule with three different inclination angles ($\theta_0=0$, $\theta_0=\pi/4$, and $\theta_0=\pi/2$) has been simulated under simple shear. In Fig. 9 (right), the steady interface configurations corresponding to these three initial inclination angles are presented. The corresponding initial configurations of capsules are shown in Fig. 9 (left). It is observed that the capsule deforms into the same steady shape for three inclinations, and this shows the existence of a unique steady state, i.e., the steady deformed capsule shapes are independent of the initial inclination angle. Among the three computations with different initial inclination angles, the agreement on the steady state tank-treading velocity is fairly excellent. Also good agreement with [41] is found. Fig. 10 shows the temporal evolution of the inclination angle of capsules with different initial inclination

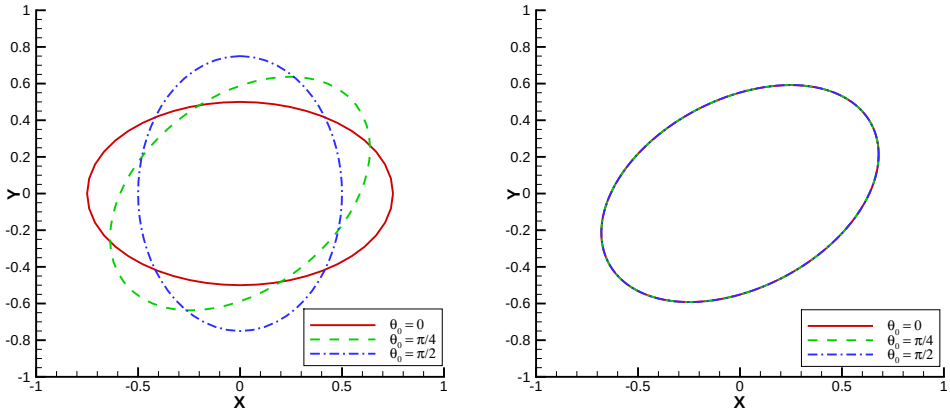

Figure 9: For Example 6.1. Initial and final shape of capsules with different initial incidences.

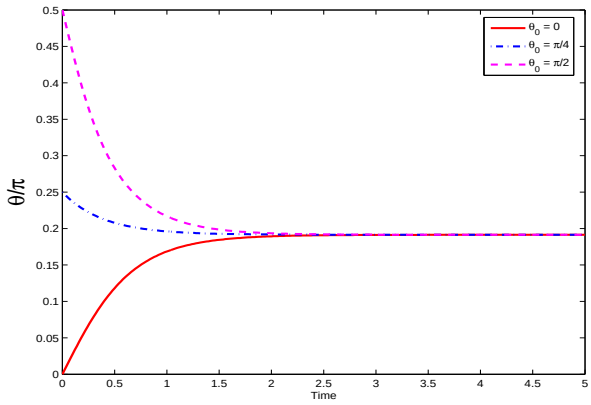

Figure 10: For Example 6.1. Temporal evolution of orientation angle of capsules with different initial incidences.

angles, respectively. The angle of inclination is computed by the approach of Breyiannis et al. [3], which involves mapping a deformed capsule to an ellipse that shares the tensor of the moments of inertia. The principal direction of this tensor is identified with the capsule inclination. It is seen that the time the capsule takes to reach steady state is different for different initial inclination angles, longest for $\theta_0 = 0$ while shortest for $\theta_0 = \pi/4$.

Fig. 11 shows the initial and steady shape of capsules with different capsule circularity $c = \frac{1}{2\sqrt{\pi}} L A^{-\frac{1}{2}}$, where L and A is the length and area of the capsule, respectively. It is observed that the steady shape of capsule with high circularity is significantly different the initial elliptical shape shown as the dash-dotted and dotted lines in the figure. Fig. 11(right) shows that the larger the circularity of the capsule, the smaller the inclination angle of capsule at steady state. This is verified more clearly in Fig. 12 which shows the steady inclination angle in terms of the circularity. These are in good agreement with previous results in the literature [33, 41].

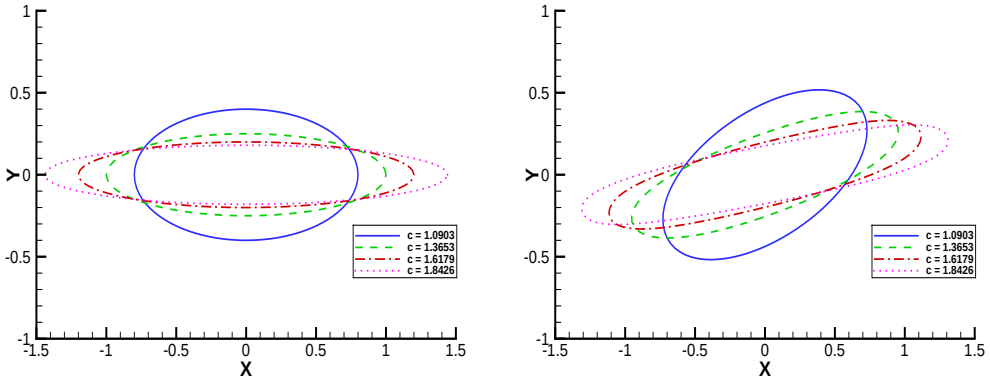

Figure 11: For Example 6.1. Initial and steady shapes of capsules corresponding to different the values of capsule circularity.

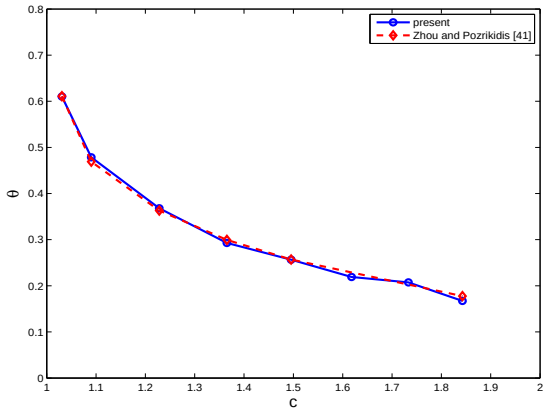

Figure 12: For Example 6.1. Comparison of the orientation angle of steady capsules versus circularity with that obtained by Zhou and Pozrikidis [41]

In Fig. 13 (left) and Fig. 13 (right), the shapes of capsules with different shear rates at $t=0.059$ and at steady state are plotted, respectively, which shows the effects of different shear rates on the deformation of capsules. Fig. 14 shows that the inclination angle of steady capsule is almost independent of the shear rate as in [33].

In Table 1, the convergence rate analysis for the case of $\theta_0=0$ at $t=0.01$ is shown, using a reference grid of 1024×516, and the expected second order accuracy for the velocity and near second order accuracy for the pressure are observed.

Table 1: Grid refinement analysis for Example 6.1 at $t=0.01$.

$M\times N$	$\|E_{\mathbf{u}}\|_\infty$	Order	$\|E_p\|_\infty$	Order
128×64	2.3866E-04	–	3.6558E-03	–
256×128	5.3954E-05	2.15	1.0572E-03	1.79
512×256	1.2797E-05	2.08	3.2238E-04	1.71

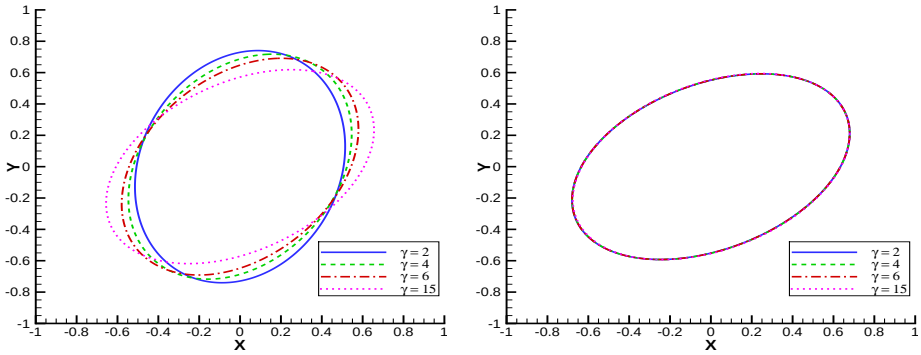

Figure 13: For Example 6.1. The shapes of capsules with different shear rates at $t=0.059$ (left) and at steady state (right).

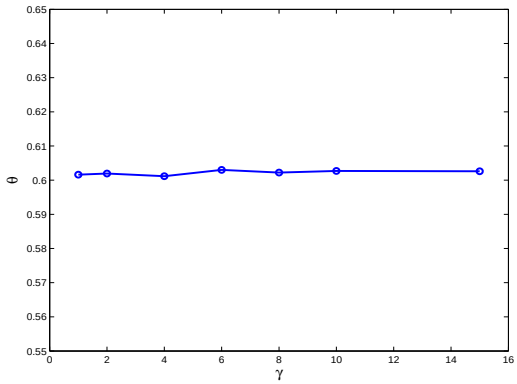

Figure 14: For Example 6.1. The orientation angle of steady capsules versus shear rates.

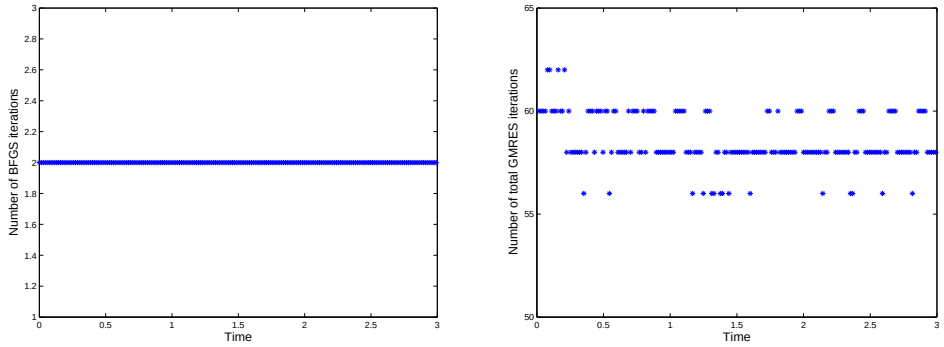

Figure 15: Number of BFGS iterations (left) and Number of total GMRES iterations (right) versus time with $\theta_0=0$.

In Fig. 15(left) and Fig. 15(right), the plots of the number of BFGS iterations and the number of total GMRES iterations versus time with $\theta_0=0$ are presently, respectively. The stop tolerances for the BFGS and the GMRES method are 10^{-9} and 10^{-6}, respectively. It can be seen that the number of BFGS iterations is few and the total GMRES iterations

is not very large for the present case. Here, the total GMRES iterations means the sum of the number of GMRES iteration at each BFGS iteration within one time step, and the number of average GMRES iterations is about 30 at each BFGS iteration. It takes about 14 iterations for the CG to converge with a tolerance of 10^{-8} at each GMRES iteration. Above indicates that a limited number of iterations are needed.

Example 6.2. An initially biconcave capsule

In this example, the deformation of an initially biconcave capsule in shear flow is simulated to show the ability of the current method for handling a more complex geometry problem. The shape of capsule is given by

$$X(\psi)=\sin(\psi), \tag{6.1}$$
$$Y(\psi)=0.5(E_0+E_1\sin^2\psi+E_2\sin^4\psi)\cos\psi, \tag{6.2}$$

where $E_0=0.207161$, $E_1=2.002558$, $E_2=-1.122762$, and ψ is a parameter which takes values within the range from 0 to 2π, see [7, 41]. The shear rate taken is $\gamma=1$. The computational domain is $[-3,3]\times[-1.5,1.5]$, and the capsule is located at the center of the domain. In the simulations, a 128×64 grid is used, and 96 control points are used to represent the interface. The fluid viscosity is set as 0.1 and the time step is taken as 10^{-3} in the computations.

First, the biconcave capsule is aligned with shear flow, i.e., the initial orientation angle of the capsule with respect to the x-axis is $\theta_0=0$. Fig. 16 shows the shapes and positions of deformed capsule at a time series of $t=0.5$, $t=2.0$, $t=3.0$ and $t=9.0$. It is found that the capsule has large deformation compared to the previous first example. In the beginning, the deformation leads to formation of two symmetric pockets originating at the dimples. As the cell deforms, the depth of the pockets is flattened with their depth reduced, while the depth at the middle of the capsule is increased. The capsule finally achieves a smooth,

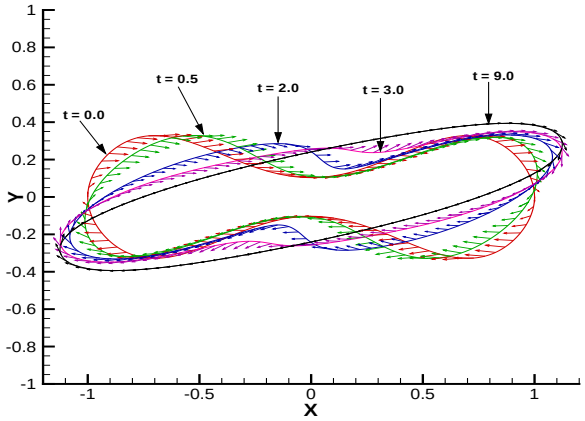

Figure 16: For Example 6.2. Shapes of deformed capsules at different times. The velocities at the control points of inextensible interfaces are shown as the arrows.

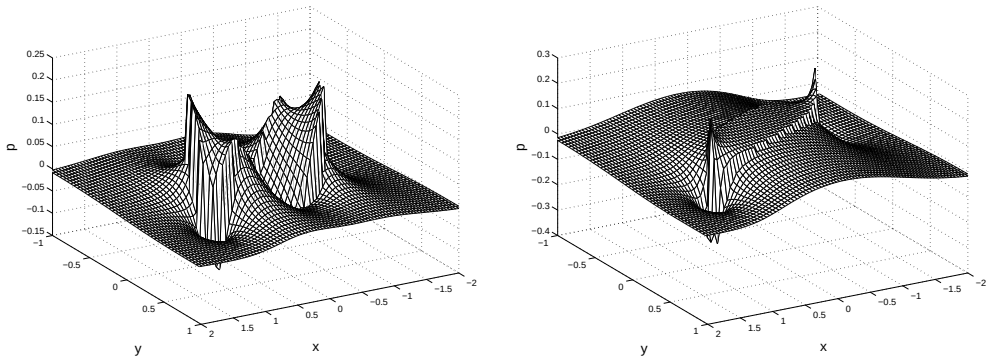

Figure 17: For Example 6.2. Pressure profiles at $t=0.5$ (left) and $t=7.5$ (right) around the capsule.

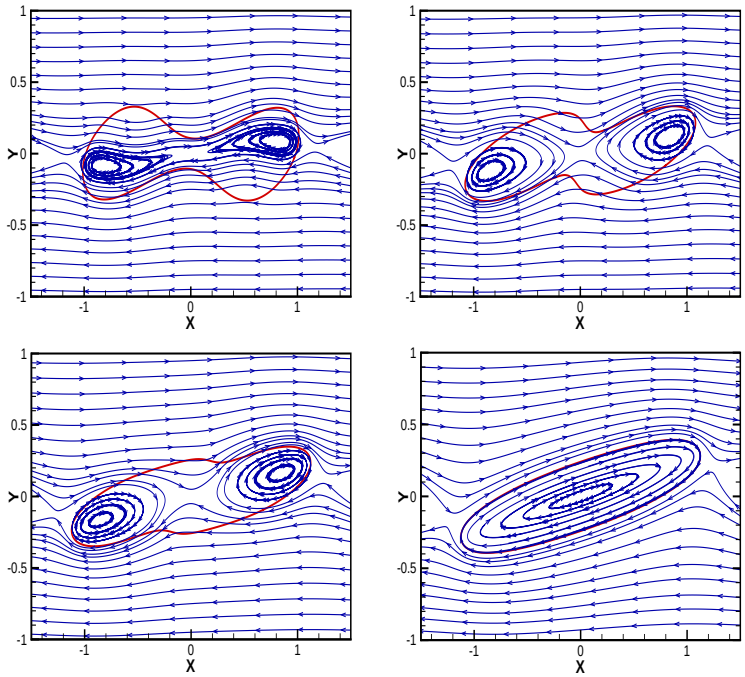

Figure 18: For Example 6.2. Streamline pattern at different times, $t=0.5$, $t=2.0$, $t=3.0$, and $t=9.0$ (from top to bottom, from left to right, respectively).

elongated, narrow almost elliptical steady shape. After that, the capsule then rotates around the liquid inside with a nearly unchanged shape and presents a tank-treading motion. The pressure profiles at $t=0.5$ and $t=7.5$ are presented in Fig. 17 (left) and Fig. 17 (right), respectively. Again it is observed that the sharp jump in pressure across the interface is well resolved by the present method.

The plots of streamlines at time $t=0.5$, $t=2.0$, $t=3.0$ and $t=9.0$ are presented in Fig. 18, where the bold closed solid line represents the configuration of the capsule. It is

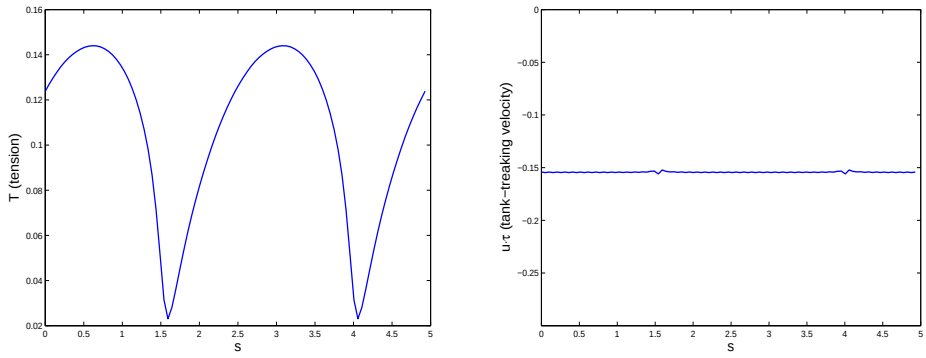

Figure 19: For Example 6.2. The interface tension (left) and tank-treading velocity (right) versus arc length at steady state.

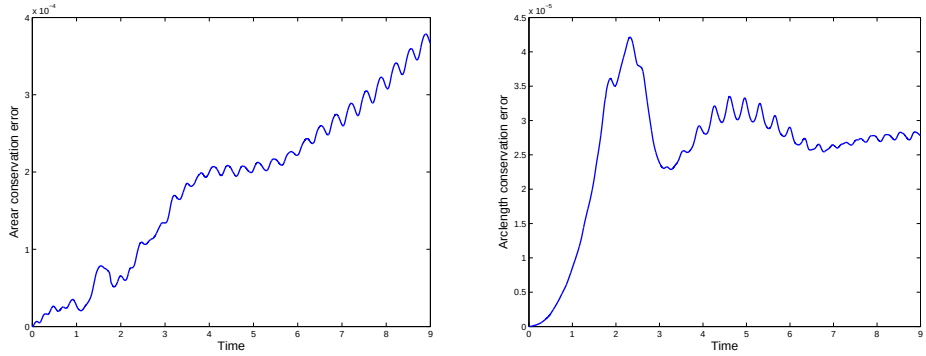

Figure 20: For Example 6.2. The area conservation error (left) and arclength conservation error (right) versus time.

seen that the two internal eddies are formed inside the capsule and later the two vortices have coalesced into a single vortex and the recirculating regions are observed. At $t=9.0$, it is observed that a closed streamline has been almost aligned with the capsule interface, which indicates a steady-state shape of capsule is achieved at this moment. The capsule with large deformation rotates around the liquid inside and the fluid flow inside the capsule moves in the clockwise direction. Compared with those in the first example as shown in Fig. 6, the flow patterns around the capsule has the significant changes.

The distribution of the interface tension and tank-treading velocity versus arclength at steady state are shown in Fig. 19 (left) and Fig. 19 (right), respectively. Again it is observed from Fig. 19 (right) that tank-treading velocity remains almost constant along the interface of the capsule due to the interface inextensibility. The capsule engages in a tank-treading motion around the capsule with uniform tangential velocity at the steady state. The distribution of the tension versus arclength with significant variations is presented in Fig. 19 (left). As in the case of initially elliptical capsules, the tension is minimum near the point of maximum interface curvature and while maximum near the point of minimum interface curvature. The results are in agreement with [41].

Fig. 20 (left) shows the plot of the area conservation error versus time up to $t=5$. In the figure, the maximum error in the area is less than $4.5e-004$. The plot of arclength conservation error versus time is presented in Fig. 20 (right), and the maximum error in the capsule arclength is less than $4.5e-05$. These figures shows that the area enclosed by the capsule and arclength of capsule are both conserved well. Compared to the first example, note that the losses of the area and arclength are relatively larger, the possible reason is that the capsule is elongated due to the large deformation and therefore results in the coarser distribution of the control points in some parts.

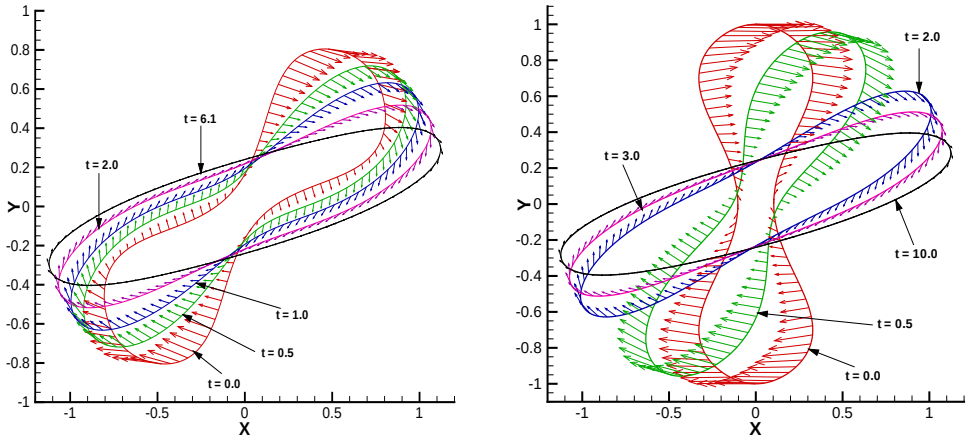

Figure 21: For Example 6.2. Shapes of deformed capsules at different times with $\theta_0=\pi/4$ (left) and with $\theta_0=\pi/2$ (right). The velocities at the control points of inextensible interfaces are shown as the arrows.

Fig. 21 (left) and Fig. 21 (right) show a sequence of evolving shapes of deformed capsules at different times with different initial incidences, $\theta_0=\pi/4$ and $\theta_0=\pi/2$, respectively. It is seen from these figures that the capsule undergoes a large deformation and finally reaches a steady shape, which is compared to Fig. 16. The corresponding streamlines at different times with $\theta_0=\pi/4$ and $\theta_0=\pi/2$ are presented in Fig. 22 and Fig. 23, respectively. The significantly different flow patterns are observed before the steady state is achieved, which can be also contrasted to those found in Fig. 18. At the steady state, the capsule rotates around the liquid inside. The flow patterns and shape of capsule are almost the same at the moment, which indicates that the steady shape and orientation of the capsule are independent of the initial inclination angle.

The deformation of the capsule is described by the Taylor shape parameter D_{xy} defined as $(L-B)/(L+B)$, where L and B are the maximum and minimum radial distances of a cross-section of the cylindrical capsule, respectively. The temporal evolution of the Taylor deformation parameter and inclination angle of capsules with different initial incidences are presented in Fig. 24 (left) and Fig. 24 (right). The amplitude in the deformation parameter with $\theta_0=0$ first decreases and then increases, which is distinct from those with $\theta_0=\pi/4$ and $\theta_0=\pi/2$. The same orientation angle of capsule at the steady state is observed as shown in Fig. 24 (right).

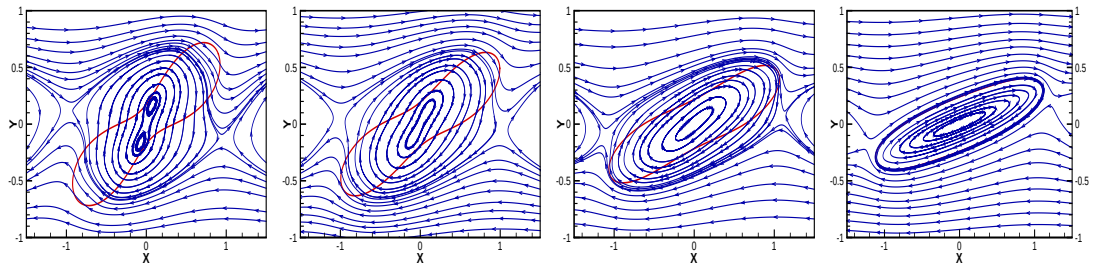

Figure 22: For Example 6.2. Streamline pattern at different times with $\theta_0=\pi/4$, $t=0.5$, $t=1.0$, $t=2.0$, and $t=6.1$ (from left to right).

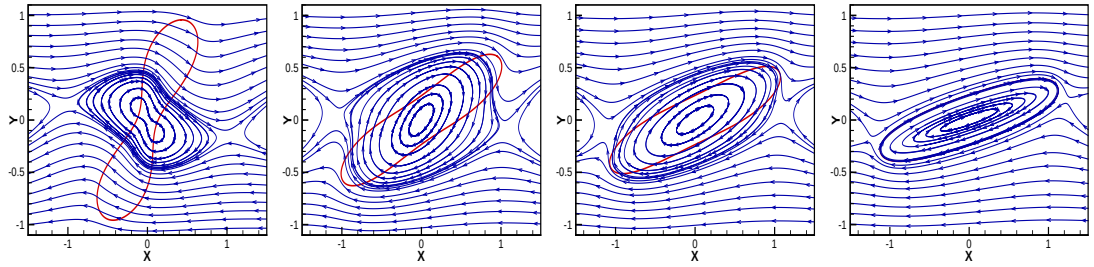

Figure 23: For Example 6.2. Streamline pattern at different times with $\theta_0=\pi/2$, $t=0.5$, $t=2.0$, $t=3.0$, and $t=10.0$ (from left to right).

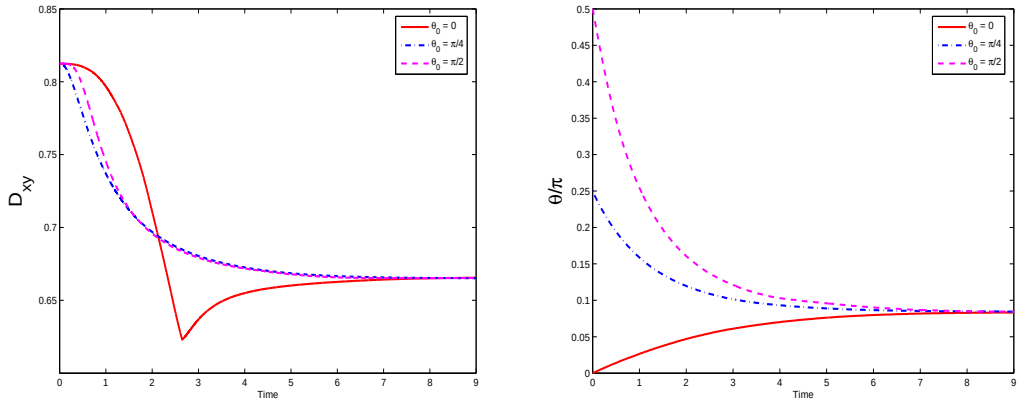

Figure 24: For Example 6.2. The Taylor deformation parameter (left) and inclination angle of capsules (right) versus time with different initial incidences.

Finally, the effect of the bending force is tested by two cases. The first test case is in presence of fluid flow. Fig. 25 shows the deformation and shape of an initially elliptical capsule with $\theta_0=\frac{3}{4}\pi$ at shear rate $\gamma=2$. It can be seen from this figure that the interface with smaller bending coefficient has larger deformation in some part of the capsule before reaching the steady state, however there is no significant influence on the final steady shape for this case.

For the second test case, a capsule suspended freely in a quiescent flow is consid-

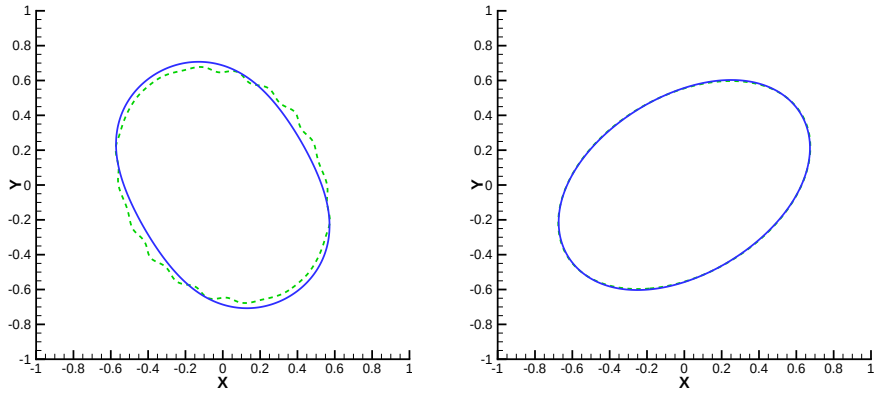

Figure 25: For Example 6.1. The shape of capsules with different bending coefficients at $t=0.199$ (left) and $t=0.999$ (right), $c_b=0.01$ (solid line) and $c_b=0.0001$ (dashed line).

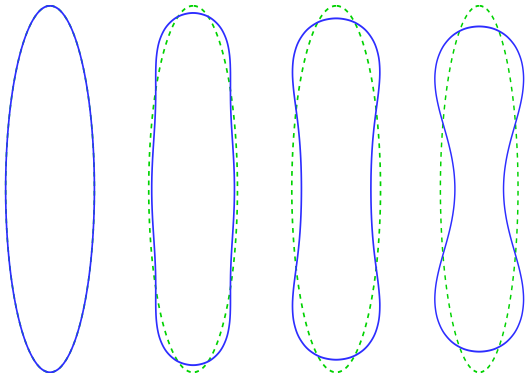

Figure 26: For Example 6.1. The shape of capsules with different bending coefficients at different times $t=0$, $t=2.999$, $t=9.999$ and at steady state (left to right, respectively). The circularity is taken as 1.3653, $c_b=10^{-3}$ (solid line) and $c_b=10^{-7}$ (dashed line).

ered. In the absence of inextensibility constraint, the steady shape of a capsule is a circle. However, the steady shape can be different from a circle since the capsule is locally inextensible. In [33], it is shown that the steady shape depends only on the reduced area of the capsule and is independent of the material properties of the capsule and surrounding fluid. In Fig. 26, the motion of ellipse-shaped capsule with two different bending coefficients at different times is shown. The ellipse with a smaller bending rigidity has the same initial shape as shown in the dashed line, however the ellipse with a larger bending rigidity relaxes to a biconcave shape as shown in the solid line.

7 Concluding remarks

In this paper, an immersed interface method for solving steady Stokes flows with inextensible interfaces on a MAC grid is presented, which is used to simulate the deformation

of liquid capsules enclosed by inextensible interfaces in simple shear flow. The method combines the immersed interface method with a front tracking representation of the interface on a uniform Cartesian grid. The tension is introduced as an unknown to maintain the condition of interface inextensibility and the resulting system is solved by the GMRES method. Based on the determined tension, the forces that the inextensible interface exerts on the fluid are calculated from the location of the interface, where the location of the interface is updated implicitly within each time step. With the present method, the area enclosed by the interface and the arclength of the interface are conserved well simultaneously. The capability of the proposed method to simulate Stokes flows with inextensible interfaces are demonstrated by some numerical examples. Future works include the extension to incompressible two-phase flow involving inextensible interface. Finally, we plan to generalize the current work to 3D simulations. The present technique can be applied to 3D. In 3D, the interface is a surface and can be discretized using triangular mesh. Singular forces related to augmented variables are computed at the nodes of the triangulations, which are used to compute the jump conditions of the solutions and their derivatives.

Acknowledgments

The authors would like to thank the referees for the valuable suggestions on the revision of the manuscript. The research of the first author was partially supported by Guangdong Provincial Government of China through the "Computational Science Innovative Research Team" program, the Sun Yat-sen University "Hundred Talents Program" (34000-3181201), and the National Natural Science Foundation of China (No. 11101446).

References

[1] J. Adams, P. Swarztrauber, R. Sweet, FISHPACK: Efficient FORTRAN subprograms for the solution of separable eliptic partial differential equations, 1999. Available on the web at `http://www.scd.ucar.edu/css/software/fishpack/`.

[2] R.P. Beyer, A computational model of the cochlea using the immersed boundary method, J. Comput. Phys., 98 (1992), pp. 145–162.

[3] G. Breyiannis, C. Pozrikidis, Simple shear flow of suspensions of elastic capsules, Theor. Comp. Fluid Dyn., 13 (2000), pp. 327–47.

[4] D. Calhoun, A Cartesian grid method for solving the two-dimensional streamfunction-vorticity equations in irregular regions, J. Comput. Phys., 176 (2002), pp. 231–275.

[5] B. Christoph, Domain imbedding methods for the Stokes equations, Numerische Mathematik, 57 (1990), pp. 435–451.

[6] R. Dillon, L. Fauci, A. Fogelson, and D. Gaver, Modeling biofilm processes using the immersed boundary method, J. Comput. Phys., 129 (1996), pp. 57–73.

[7] C.D. Eggleton and A.S. Popel, Large deformation of red blood cell ghosts in a simple shear flow, Phys. Fluids, 10 (1998), pp. 1834–1845.

[8] H.C. Elman, Multigrid and Krylov subspace methods for the discrete Stokes equations, Int. J. Numer. Meth. Fluids, 227 (1996), pp. 755–770.
[9] L.J. Fauci and C.S. Peskin, A computational model of aquatic animal locomotion, J. Comput. Phys., (1988), pp. 85–108.
[10] A.L. Fogelson, A mathematical model and numerical method for studying platelet adhesion and aggregation during blood clotting, J. Comput. Phys., 1 (1984), pp. 111–134.
[11] A.L. Fogelson, Continuum models of platelet aggregation: Formulation and mechanical properties, SIAM J. Appl. Math., 52 (1992), pp. 1089–1110.
[12] D.V. Le, B.C. Khoo, and J. Peraire, An immersed interface method for viscous incompressible flows involving rigid and flexible boundaries, J. Comput. Phys., 220 (2006), pp. 109–138.
[13] L. Lee and R.J. LeVeque, An immersed interface method for incompressible Navier–Stokes equations, SIAM J. Sci. Comput., 25 (2003), pp. 832–856.
[14] R.J. LeVeque and Z. Li, Immersed interface methods for Stokes flow with elastic boundaries or surface tension, SIAM J. Sci. Comput., 18 (1997), pp. 709–735.
[15] R.J. LeVeque and Z. Li, The immersed interface method for elliptic equations with discontinuous coefficients and singular sources, SIAM J. Numer. Anal., 31 (1994), pp. 1019–1044.
[16] Z. Li and K. Ito, The immersed interface method-numerical solutions of PDEs involving interfaces and irregular domains. SIAM Frontiers Appl. Math., (2006), pp. 33.
[17] Z. Li and M.C. Lai, The immersed interface method for the Navier-Stokes equations with singular forces, J. Comput. Phys., 171 (2001), pp. 822–842.
[18] Z. Li and C. Wang, A fast finite difference method for solving Navier-Stokes equations on irregular domains, Comm. Math. Sci., 1 (2003), pp. 180–196.
[19] M.N. Linnick and H.F. Fasel, A high-order immersed interface method for simulating unsteady incompressible flows on irregular domains, J. Comput. Phys., 204 (2005), pp. 157–192.
[20] C.W. Oosterlee and F.J.G. Lorenz, Multigrid methods for the Stokes system, Comput. Sci. Engr., 8 (2006), pp. 34–43.
[21] C.S. Peskin, Numerical analysis of blood flow in the heart, J Comput. Phys., 25 (1977), pp. 220–52.
[22] C.S. Peskin, The immersed boundary method, Acta Numerica, 11 (2002), pp. 479–517.
[23] J. Peters, V. Reichelt, and A. Reusken, Fast iterative solvers for discrete Stokes equations, SIAM J. Sci. Comput., 27 (2005), pp. 646–666.
[24] C. Pozrikidis, The axisymmetric deformation of a red blood cell in uniaxial straining Stokes flow, J. Fluid Mech. 216 (1990), pp. 231–254.
[25] J.M. Stockie and S.I. Green, Simulating the motion of flexible pulp fibres using the immersed boundary method, J. Comput. Phys., 147 (1998), pp. 147–165.
[26] D. Russell and Z.J. Wang, A Cartesian grid method for modeling multiple moving objects in 2D incompressible viscous flow, J. Comput. Phys., 191 (2003), pp. 177–205.
[27] Y. Sadd, GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM J. Sci. Stat. Comput., 7 (1986), pp. 856–869.
[28] U. Seifert, Configurations of fluid membranes and vesicles, Adv. Phys., 46 (1997), pp. 13–137.
[29] D. Shin and J.C. Strikwerda, Fast solvers for finite difference approximations for the Stokes and Navier–Stokes equations, J. Australian Math. Soc., 38 (1996), pp. 274–290.
[30] M.J. Stevens, Coarse-grained simulations of lipid bilayers, J. Chem. Phys., 121 (2004), pp. 11942–11948.
[31] J. Stoer and R. Bulirsch, Introduction to Numerical Analysis, 3rd ed., Springer-Verlag, 2002.
[32] E.Y. Tau, Numerical solution of the steady Stokes equations, J. Comput. Phys., 99 (1992), pp. 190–195.

[33] S.K. Veerapaneni, D. Gueyffier, D. Zorin, and G. Biros, A boundary integral method for simulating the dynamics of inextensible vesicles suspended in a viscous fluid in 2D, J. Comput. Phys., 228 (2009), pp. 2334–2353.
[34] S.K. Veerapaneni, D. Gueyffier, G. Biros, and D. Zorin, A numerical method for simulating the dynamics of 3D axisymmetric vesicles suspended in viscous flows, J. Comput. Phys., 228 (2009), pp. 7233-7249.
[35] Z.-J. Tan, D.V. Le, Z. Li, K.M. Lim and B.C. Khoo, An immersed interface method for solving incompressible viscous flows with piecewise constant viscosity across a moving elastic membrane, J. Comput. Phys., 227 (2008), pp. 9955–9983.
[36] Z.-J. Tan, K.M. Lim and B.C. Khoo, A fast immersed interface method for solving Stokes flows on irregular domains, Comput. Fluids, 38 (2009), pp. 1973–1983.
[37] N.T. Wang and A.L. Fogelson, Computational methods for continuum models of platelet aggregation, J. Comput. Phys, 151 (1999), pp. 649–675.
[38] A. Wiegmann and K.P. Bube, The explicit-jump immersed interface method: Finite difference methods for PDEs with piecewise smooth solutions, SIAM J. Numer. Anal., 37 (2000), pp. 827–862.
[39] S. Xu and Z.J. Wang, An immersed interface method for simulating the interaction of a fluid with moving boundaries, J. Comput. Phys., 216 (2006), pp. 454–493.
[40] S. Xu and Z.J. Wang, A 3D Immersed Interface Method for Fluid-Solid Interaction, Comput. Methods Appl. Mech. Engr., 197 (2008), pp. 2068–2086.
[41] H. Zhou and C. Pozrikidis, Deformation of liquid capsules with incompressible interfaces in simple shear flow, J. Fluid Mech., 283 (1995), pp. 175–200.