中文摘要

在這份報告中，我們利用分項步驟法去解 Stokes 流體中不可壓縮膜的問題。我們將整組偏微分方程離散，並順利得到有關速度、壓力及表面張力的線性方程組。所得到的矩陣是對稱的，而且我們能控制其局部不可壓縮的特性，所得數值結果與文獻吻合。

關鍵詞:內嵌邊界法，不可壓縮膜，分項步驟法，Stokes 流體

A fractional step immersed boundary method for Stokes flow with an inextensible interface

Ming-Chih Lai,* Wei-Fan Hu,† and Wen-Wei Lin‡

Abstract

In this paper, we develop a fractional step method based on the immersed boundary (IB) formulation for Stokes flow with an inextensible (incompressible) interface. In addition to solving the fluid variables such as the velocity and pressure, the present problem involves finding extra unknown elastic tension such that the surface divergence of the velocity is zero along the interface. The equations are discretized by standard centered difference schemes on a staggered grid and the interactions between the interface and fluids are discretized using the discrete delta function as in the immersed boundary method. The resulting linear system of equations is symmetric and can be solved by fractional steps so that only fast Poisson solvers are involved. The present method can be extended to Navier-Stokes flow without any difficulty by treating the nonlinear advection terms explicitly during the time evolution. The convergence tests for Stokes solver with or without an inextensible interface are performed and confirm the desired accuracy. The tank-treading motion for an inextensible interface under a simple shear flow has been studied extensively and the results are in good agreement with those obtained in literature.

Key words: Immersed boundary method; Inextensible interface; Fractional step method; Stokes flow

AMS subject classifications: 65N06, 76D07, 76D05

1 Introduction

In this paper, we develop a fractional step method based on immersed boundary (IB) formulation for Stokes flow with an inextensible (incompressible) interface. In addition to solving the fluid variables such as the velocity and pressure, the present problem involves finding extra unknown elastic tension such that the surface divergence of the velocity is zero along the interface. Once the velocity is found, the interface moves with the local fluid velocity as usual. Since the fluid is incompressible and the interface is inextensible, both

*Corresponding author. Center of Mathematical Modeling and Scientific Computing & Department of Applied Mathematics, National Chiao Tung University, 1001, Ta Hsueh Road, Hsinchu 300, Taiwan. mclai@math.nctu.edu.tw

†Department of Applied Mathematics, National Chiao Tung University, 1001, Ta Hsueh Road, Hsinchu 300, Taiwan. weifanhu.am95g@g2.nctu.edu.tw

‡Department of Mathematics, National Taiwan University, No. 1, Sec. 4, Roosevelt Road, Taipei, Taiwan. wwlin@math.ntu.edu.tw.

the area enclosed by the interface and the total length of the interface should be conserved. The mathematical model is motivated by the simulation of the vesicle dynamics [9], the deformation of erythrocytes [13, 15] and drug-carrying capsules [18], just to name a few. In particular, the dynamics of moving vesicles has been extensively explored both experimentally [5, 8] (and the references therein) and computationally [9, 24, 22, 19, 6, 10]. Notice that, the dynamics of vesicles are determined by their boundary rigidity, inextensibility, and the hydrodynamical forces. Our present model for inextensible interface can be regarded as a simplified vesicle model without bending effect.

The rest of the paper are organized as follows. In the next section, we describe our governing equations for the Stokes flow with an inextensible interface based on immersed boundary formulation. We also show the skew-adjoint property between the spreading operator acting on the tension and the surface divergence of the velocity. In Section 3, the symmetry of the resulting matrix equation is provided first, and then a numerical algorithm based on the fractional step method is developed. Finally, the numerical results including the convergence tests and the tank-treading motion for an inextensible interface under a simple shear flow are shown in detail in Section 4.

2 Governing equations

We begin by stating the mathematical formulation of the Stokes flow with an inextensible interface. Consider the existence of a moving, immersed, inextensible interface $\Gamma(t)$ in the fixed fluid domain Ω. We assume that the fluids inside and outside of the interface are the same so the governing equations in immersed boundary formulation can be written as follows [6].

$$-\nabla p + \mu\Delta\mathbf{u} + \int_\Gamma \frac{\partial}{\partial s}(\sigma\boldsymbol{\tau})\delta(\mathbf{x}-\mathbf{X}(s,t))\,\mathrm{d}s = 0, \quad \text{in } \Omega, \tag{1}$$

$$\nabla\cdot\mathbf{u} = 0, \quad \text{in } \Omega, \tag{2}$$

$$\nabla_s\cdot\mathbf{U} = 0, \quad \text{on } \Gamma, \qquad \mathbf{U}(s,t) = \int_\Omega \mathbf{u}(\mathbf{x},t)\delta(\mathbf{x}-\mathbf{X}(s,t))\,\mathrm{d}\mathbf{x}. \tag{3}$$

$$\frac{\partial\mathbf{X}}{\partial t}(s,t) = \mathbf{U}(s,t), \quad \text{on } \Gamma \tag{4}$$

Here μ is a constant fluid viscosity, $\mathbf{u} = (u, v)$ and p are the velocity field and the pressure both described in Cartesian coordinates. The interface Γ is represented by a parametric form $\mathbf{X}(s,t) = (X(s,t), Y(s,t))$, where s is the Lagrangian parameter of the initial configuration.

Eqs. (1) and (2) are the familiar incompressible Stokes equations with a singular force term arising from the interface. Eq. (3) represents the inextensibility constraint of the interface which is equivalent to the zero surface divergence (defined later) of the velocity along the interface. This inextensible constraint is due to the fact that the local stretching factor [12]

$$\frac{\partial}{\partial t}\left|\frac{\partial\mathbf{X}}{\partial s}\right| = (\nabla_s\cdot\mathbf{U})\left|\frac{\partial\mathbf{X}}{\partial s}\right| \quad \text{on } \Gamma$$

must be zero along the interface. Here, the interfacial velocity $\mathbf{U}$ is simply the interpolation of the fluid velocity at the interface which is defined as in the traditional IB formulation. Eq. (4) simply represents that the interface moves along with the local fluid velocity (the interfacial velocity). The interaction between the fluid and the interface is linked by the two-dimensional Dirac delta function $\delta(\mathbf{x}) = \delta(x)\delta(y)$. Unlike the traditional IB formulation in which the elastic tension $\sigma(s,t)$ is either known or a function of immersed boundary configurations, here, the tension is a part of solution needed to be determined. In fact, the tension (defined on the interface) plays the role of the Lagrange multiplier to enforce the local inextensible constraint Eq. (3) of the interface which is exactly like the role played by the pressure to enforce the incompressible constraint Eq. (2) of the fluid. In this model, we consider the Stokes flow; however, the numerical method can be extended to Navier-Stokes flow straightforwardly by treating the nonlinear advection terms explicitly during the time evolution.

The difficulties in solving the above interfacial problem are as follows. Firstly, since the fluid is incompressible and the interface is inextensible, both the area enclosed by the interface and total length of the interface should be conserved simultaneously. Furthermore, the local inextensibility constraint (3) is more stringent than the conservation of the total interfacial length since the latter is a global constraint. Secondly, the elastic tension must be treated as an unknown function which needs to be solved with the fluid variables simultaneously. In the previous literature, most of the related work is based on boundary integral methods, see for example, [24, 22, 19] and the references therein. However, boundary integral methods generally assume infinite domains, and cannot be generalized to full Navier-Stokes equations since there is no corresponding Green function. Until recently, Kim and Lai [6] have applied a penalty immersed boundary method to simulate the dynamics of inextensible vesicles. By introducing two different kinds of Lagrangian markers, the authors are able to decouple the fluid and vesicle dynamics so that the computation can be performed more efficiently. One potential problem of this approach is that the time step depends on the penalty number and must be chosen smaller as the penalty number becomes larger. In [10], a new finite difference scheme based on the immersed interface method has been developed for solving the present problem in Navier-Stokes flow. The authors treat the unknown elastic tension as an augmented variable so that the augmented IIM can be applied. In this paper, we discretize the equations (1)-(3) directly without decoupling and use a fractional step method to solve the resulting linear system of equations. The numerical algorithm will be given in next section.

Before continuing, we first prove that the spreading operator acting on the function σ and the surface divergence operator of the velocity are skew-adjoint with each other. To proceed, let us define the spreading operator S of σ and the surface divergence operator ∇_s of $\mathbf{U}$ as follows.

$$S(\sigma) = \int_\Gamma \frac{\partial}{\partial s}(\sigma\boldsymbol{\tau})\delta(\mathbf{x} - \mathbf{X}(s))\ \mathrm{d}s. \tag{5}$$

$$\nabla_s \cdot \mathbf{U} = \frac{\partial \mathbf{U}}{\partial \boldsymbol{\tau}} \cdot \boldsymbol{\tau} = \frac{\partial \mathbf{U}}{\partial s} \cdot \boldsymbol{\tau} / \left|\frac{\partial \mathbf{X}}{\partial s}\right| = \frac{\partial}{\partial s}\left(\int_\Omega \mathbf{u}(\mathbf{x})\delta(\mathbf{x} - \mathbf{X}(s))\ \mathrm{d}\mathbf{x}\right) \cdot \boldsymbol{\tau} / \left|\frac{\partial \mathbf{X}}{\partial s}\right|. \tag{6}$$

We also define the inner product of functions on Ω and Γ in the following.

$$\langle \mathbf{u}, \mathbf{v} \rangle_\Omega = \int_\Omega \mathbf{u}(\mathbf{x}) \cdot \mathbf{v}(\mathbf{x}) \, \mathrm{d}\mathbf{x}, \quad \langle f, g \rangle_\Gamma = \int_\Gamma f(l) \, g(l) \, dl, \tag{7}$$

where l in Eq. (7) is the arc-length parameter. Then we have

$$\begin{aligned}
\langle S(\sigma), \mathbf{u} \rangle_\Omega &= \int_\Omega \left(\int_\Gamma \frac{\partial}{\partial s}(\sigma \boldsymbol{\tau}) \delta(\mathbf{x} - \mathbf{X}(s)) \, \mathrm{d}s \right) \cdot \mathbf{u}(\mathbf{x}) \, \mathrm{d}\mathbf{x} \\
&= \int_\Gamma \frac{\partial}{\partial s}(\sigma \boldsymbol{\tau}) \cdot \left(\int_\Omega \mathbf{u}(\mathbf{x}) \delta(\mathbf{x} - \mathbf{X}(s)) \, \mathrm{d}\mathbf{x} \right) \, \mathrm{d}s \\
&= - \int_\Gamma \sigma \left(\boldsymbol{\tau} \cdot \frac{\partial \mathbf{U}}{\partial s} \right) \, \mathrm{d}s \quad \text{(intergartion by parts and the closed interface)} \\
&= \int_\Gamma \sigma \left(-\frac{\partial \mathbf{U}}{\partial s} \cdot \boldsymbol{\tau} / \left| \frac{\partial \mathbf{X}}{\partial s} \right| \right) \left| \frac{\partial \mathbf{X}}{\partial s} \right| \, \mathrm{d}s \\
&= \langle \sigma, -\nabla_s \cdot \mathbf{U} \rangle_\Gamma = \langle \sigma, S^*(\mathbf{U}) \rangle_\Gamma
\end{aligned} \tag{8}$$

By comparison, we conclude that the spreading operator and the surface divergence operator are skew-adjoint.

The reason for showing the skew-adjointness of those two operators are two fold. Firstly, since the surface divergence of velocity is zero in Eq. (8), from the above derivation, we have $\langle S(\sigma), \mathbf{u} \rangle_\Omega = 0$. That is, the present elastic tension does not do work to the fluid, which is not surprising since it is the Lagrange multiplier for the inextensible constraint. However, if we add the bending force along the interface as the one in vesicle problems [22, 6], then the bending force does do work to the fluid. Secondly, the skew-adjointness is also satisfied in the discrete sense (see in next section) so that the resulting matrix equation is symmetric. Consequently, some efficient iterative solvers such as the conjugate gradient (CG) method can be easily applied.

3 Numerical algorithm

We now are ready to discretize Eq. (1)-(4) by the IB method. For simplicity, we assume the computational domain $\Omega = [a, b] \times [c, d]$ is a rectangular domain. The fluid variables are defined on the staggered grid introduced by Harlow and Welsh [3]; that is, the pressure is defined on the grid points labelled as $\mathbf{x} = (x_i, y_j) = (a + (i - 1/2)\Delta x, c + (j - 1/2)\Delta y)$, $i = 1, 2, \cdots, m$ and $j = 1, 2, \cdots, n$, while the velocity components u and v are defined at $(x_{i-1/2}, y_j) = (a+(i-1)\Delta x, c+(j-1/2)\Delta y)$ and $(x_i, y_{j-1/2}) = (a+(i-1/2)\Delta x, c+(j-1)\Delta y)$, respectively. Here, we assume a uniform mesh width $h = \Delta x = \Delta y$, although it is not necessary. For the immersed interface, we use a collection of discrete points $s_k = k\Delta s, k = 0, 1, \cdots M$ with the interface mesh width Δs is roughly chosen as one half of the fluid mesh h. The Lagrangian markers are denoted by $\mathbf{X}_k = \mathbf{X}(s_k) = (X_k, Y_k)$. We also assume the interface is closed so that we have $\mathbf{X}_0 = \mathbf{X}_M$. The elastic tension are defined at the "half-integered" points given by $s_{k-1/2} = (k - 1/2)\Delta s$ so we denote it as $\sigma_{k-1/2}$. Without loss of generality, for any function defined on the interface $\phi(s)$, we approximate the partial

derivative $\frac{\partial \phi}{\partial s}$ by the central difference scheme as

$$D_s\phi = \frac{\phi(s+\Delta s/2) - \phi(s-\Delta s/2)}{\Delta s}. \tag{9}$$

Thus, the interface stretching factor and the unit tangent can be approximated by $|D_s\mathbf{X}|$, and $\boldsymbol{\tau} = D_s\mathbf{X}/\left|D_s\mathbf{X}\right|$ which in turn are also defined at the "half-integered" points. We denote them by $|D_s\mathbf{X}|_{k-1/2}$ and $\boldsymbol{\tau}_{k-1/2}$, respectively.

Let Δt be the time step size, and the superscript of the variables denote the time step index. At the beginning of each time step n, the interface configuration $\mathbf{X}_k^n$, and the unit tangent $\boldsymbol{\tau}_{k-1/2}^n$ are all given. The numerical algorithm can be written as follows.

$$-\nabla_h p^{n+1} + \mu\Delta_h \mathbf{u}^{n+1} + \sum_{k=0}^{M-1} D_s\left(\sigma^{n+1}\boldsymbol{\tau}^n\right)_k \delta_h(\mathbf{x} - \mathbf{X}_k^n)\Delta s = 0, \tag{10}$$

$$\nabla_h \cdot \mathbf{u}^{n+1} = 0, \tag{11}$$

$$\nabla_{s_h} \cdot \mathbf{U}_k^{n+1} = \frac{\mathbf{U}_k^{n+1} - \mathbf{U}_{k-1}^{n+1}}{\Delta s} \cdot \boldsymbol{\tau}_{k-1/2}^n / \left|D_s\mathbf{X}^n\right|_{k-1/2} = 0, \quad \text{for} \quad k = 1, 2, \cdots, M, \tag{12}$$

$$\mathbf{U}_k^{n+1} = \sum_{\mathbf{x}} \mathbf{u}(\mathbf{x})^{n+1}\delta_h(\mathbf{x} - \mathbf{X}_k^n)\, h^2, \tag{13}$$

where the spatial operators ∇_h, Δ_h, and $\nabla_{s_h}\cdot$ are the standard second-order central difference approximations to the gradient, Laplacian, and surface divergence. Here, δ_h is a smoother version of discrete delta function developed in [23] as $\delta_h(\mathbf{x}) = \frac{1}{h^2}\phi(\frac{x}{h})\phi(\frac{y}{h})$, with

$$\phi(r) = \begin{cases} \frac{3}{8} + \frac{\pi}{32} - \frac{r^2}{4}, & |r| < 0.5, \\ \frac{1}{4} + \frac{1-|r|}{8}\sqrt{-2 + 8|r| - 4r^2} - \frac{1}{8}\arcsin\left(\sqrt{2}(|r|-1)\right), & 0.5 \le |r| \le 1.5, \\ \frac{17}{16} - \frac{\pi}{64} - \frac{3|r|}{4} + \frac{r^2}{8} + \frac{|r|-2}{16}\sqrt{-14 + 16|r| - 4r^2} & \\ \quad + \frac{1}{16}\arcsin\left(\sqrt{2}(|r|-2)\right), & 1.5 \le |r| \le 2.5, \\ 0, & 2.5 \le |r|. \end{cases}$$

One should notice that this discrete delta function has one grid wider support than the one often used in the community. The advantage of using the above smoothing delta function is to reduce oscillations of the elastic tension caused by the present IB method. One should notice that the computation for the elastic tension does appear oscillations in other literature [24, 6, 10].

Once we obtain the new velocity field $\mathbf{u}^{n+1}$ on the fluid grid, we can interpolate the new velocity to the marker points by Eq. (13), and move the Lagrangian markers to new positions. That is,

$$\mathbf{X}_k^{n+1} = \mathbf{X}_k^n + \Delta t\, \mathbf{U}_k^{n+1}. \tag{14}$$

Therefore, we have

$$\frac{\mathbf{X}_k^{n+1} - \mathbf{X}_{k-1}^{n+1}}{\Delta s} = \frac{\mathbf{X}_k^n - \mathbf{X}_{k-1}^n}{\Delta s} + \Delta t\, \frac{\mathbf{U}_k^{n+1} - \mathbf{U}_{k-1}^{n+1}}{\Delta s}.$$

By multiplying the above equation by itself and using the zero discrete surface divergence (12), we obtain the following quality

$$\left|\frac{\mathbf{X}_k^{n+1}-\mathbf{X}_{k-1}^{n+1}}{\Delta s}\right|^2=\left|\frac{\mathbf{X}_k^{n}-\mathbf{X}_{k-1}^{n}}{\Delta s}\right|^2+(\triangle t)^2\left|\frac{\mathbf{U}_k^{n+1}-\mathbf{U}_{k-1}^{n+1}}{\Delta s}\right|^2,$$

which leads to

$$|D_s\mathbf{X}^{n+1}|^2_{k-1/2}=|D_s\mathbf{X}^{n}|^2_{k-1/2}+(\triangle t)^2\,|D_s\mathbf{U}^{n+1}|^2_{k-1/2}. \tag{15}$$

Thus, we conclude the point-wise error for the local stretching factor is first-order accurate which is comparable with the accuracy of the IB method.

3.1 Discrete skew-adjoint operators

In this subsection, we show that the spreading operator S acting on the elastic tension and the surface divergence operator $\nabla_s\cdot$ acting on the velocity are also skew-adjoint in the discrete sense. That is, we shall prove the numerical identity for Eq. (8). To proceed, we first define the corresponding discrete inner product on the fluid grid Ω_h and the interfacial grid Γ_h in the following

$$\langle\mathbf{u},\mathbf{v}\rangle_{\Omega_h}=\sum_{\mathbf{x}}\mathbf{u}(\mathbf{x})\cdot\mathbf{v}(\mathbf{x})\,h^2,\qquad \langle\phi,\psi\rangle_{\Gamma_h}=\sum_{k=1}^{M}\phi_{k-1/2}\,\psi_{k-1/2}\,|D_s\mathbf{X}|_{k-1/2}\,\Delta s, \tag{16}$$

where the second summation is nothing but the mid-point rule for the second integral of (7). We also define the discrete spreading operator S_h acting on the discrete elastic tension σ_h as

$$S_h(\sigma_h)=\sum_{k=0}^{M-1}D_s\left(\sigma\boldsymbol{\tau}\right)_k\delta_h(\mathbf{x}-\mathbf{X}_k)\Delta s. \tag{17}$$

Then we have

$$\begin{aligned}
\langle S_h(\sigma_h),\mathbf{u}\rangle_{\Omega_h} &= \sum_{\mathbf{x}}\left(\sum_{k=0}^{M-1}D_s\left(\sigma\boldsymbol{\tau}\right)_k\delta_h(\mathbf{x}-\mathbf{X}_k)\Delta s\right)\cdot\mathbf{u}(\mathbf{x})\,h^2\\
&= \sum_{k=0}^{M-1}D_s\left(\sigma\boldsymbol{\tau}\right)_k\cdot\left(\sum_{\mathbf{x}}\mathbf{u}(\mathbf{x})\delta_h(\mathbf{x}-\mathbf{X}_k)\,h^2\right)\Delta s=\sum_{k=0}^{M-1}D_s\left(\sigma\boldsymbol{\tau}\right)_k\cdot\mathbf{U}_k\Delta s\\
&= \sum_{k=0}^{M-1}\frac{\sigma_{k+1/2}\,\boldsymbol{\tau}_{k+1/2}-\sigma_{k-1/2}\,\boldsymbol{\tau}_{k-1/2}}{\Delta s}\cdot\mathbf{U}_k\Delta s\\
&= -\sum_{k=1}^{M}\sigma_{k-1/2}\left(\frac{\mathbf{U}_k-\mathbf{U}_{k-1}}{\Delta s}\right)\cdot\boldsymbol{\tau}_{k-1/2}\,\Delta s\quad\text{(summation by parts)}\\
&= \langle\sigma_h,-\nabla_{s_h}\cdot\mathbf{U}\rangle_{\Gamma_h}=\langle\sigma_h,S_h^*(\mathbf{U})\rangle_{\Gamma_h}.
\end{aligned}$$

One should notice that this discrete skew-adjoint property is crucial to our immersed boundary formulation for solving Eqs. (10) to (12). Due to the fact that discrete surface divergence of the velocity is zero in Eq. (12), we can re-scale this constraint so that the resulting matrix obtained by Eq. (12) is the transpose of the resulting matrix obtained by the discrete spreading operator of the tension. One can also verify this symmetric property by expressing the terms explicitly. The detail is given in the Appendix. We now are ready to write down the linear system of equations for Eqs. (10) to (12), and develop a numerical algorithm for solving the system.

3.2 The existence of a solution

By using the staggered grid for the discretization of fluid variables, it is well-known that the matrix obtained by the discrete divergence operator of the fluid velocity can be written as the transpose of the discrete gradient operator of the pressure. As discussed in previous subsection, the resultant matrix obtained by the discrete surface divergence of velocity can be written as the transpose of the matrix obtained by the discrete spreading operator of the tension. Thus, the linear system for Eqs. (10)-(12) is symmetric and can be written as

$$\begin{bmatrix} L & G & S \\ G^T & 0 & 0 \\ S^T & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ p \\ \sigma \end{bmatrix} = \begin{bmatrix} bc_1 \\ bc_2 \\ bc_3 \end{bmatrix}, \tag{18}$$

where the sub-matrix L, G and S are represented the discrete Laplacian $\mu\Delta_h$, discrete gradient $-\nabla_h$, and the discrete spreading operator S_h. The sub-matrix size of L, G and S are $((m-1)n + m(n-1)) \times ((m-1)n + m(n-1))$, $((m-1)n + m(n-1)) \times (mn)$ and $(m-1)n + m(n-1) \times M$, respectively. The right-hand side vector $[bc_1, bc_2, bc_3]^T$ of Eq. (18) consists only of the velocity boundary conditions since the pressure does not need the boundary condition in staggered grid formulation.

Let us discuss the existence of the solution of the linear system of Eq. (18). From now on, we denote the matrix in (18) by A. As is known, without the effect of the inextensible interface, the linear system becomes pure Stokes flow as

$$\begin{bmatrix} L & G \\ G^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ p \end{bmatrix} = \begin{bmatrix} bc_1 \\ bc_2 \end{bmatrix}. \tag{19}$$

Let us denote the matrix in Eq. (19) by $\widetilde{A}$. It is also well-known that the nullity of $\widetilde{A}$ equals to one since the pressure is unique up to a constant, and the existence of a solution can be verified by using the discrete incompressible constraint (11). To be precise, since the rank of deficiency of $\widetilde{A}$ is only one, based on the algebraic structure of the sub-matrix G, the kernel of $\widetilde{A}$ is

$$\ker(\widetilde{A}) = \text{span}\{[\underbrace{0 \cdots 0}_{(m-1)n+m(n-1)} \quad \underbrace{1 \cdots 1}_{mn}]^T\}. \tag{20}$$

And for any vector $z \in \ker(\widetilde{A}^T) = \ker(\widetilde{A})$, we have

$$\begin{aligned} z^T \begin{bmatrix} bc_1 \\ bc_2 \end{bmatrix} &= \sum_{k=1}^{mn} (bc_2)_k = \sum_{j=1}^{n} \frac{u_{0,j}}{h} - \sum_{j=1}^{n} \frac{u_{m,j}}{h} + \sum_{i=1}^{m} \frac{v_{i,0}}{h} - \sum_{i=1}^{m} \frac{v_{i,n}}{h} \\ &= -\left(-\sum_{j=1}^{n} u_{0,j} h + \sum_{j=1}^{n} u_{m,j} h - \sum_{i=1}^{m} v_{i,0} h + \sum_{i=1}^{m} v_{i,n} h \right) h^{-2} \\ &= \left(\sum_{i=1}^{m} \sum_{j=1}^{n} \left(\frac{u_{i,j} - u_{i-1,j}}{h} + \frac{v_{i,j} - v_{i,j-1}}{h} \right) h^2 \right) h^{-2} \\ &= 0, \quad \text{(by the discrete incompressibility (11))} \end{aligned} \tag{21}$$

which shows the compatibility condition for the existence of a solution.

If the effect of the inextensible interface is added, the matrix $\widetilde{A}$ is augmented by S and S^T to become A as in (18). Since the matrix S comes from the discrete spreading operator of the tension, the entries of S depend on the number of moving Lagrangian markers, their positions and tangents. It is unlikely one can show rigorously that the nullity of A is exactly equal to one. However, we have checked the above statement to be true in our numerical experiments. So the apparent kernel will be

$$\ker(A) = \text{span}\{[\underbrace{0 \cdots 0}_{(m-1)n+m(n-1)} \underbrace{1 \cdots 1}_{mn} \underbrace{0 \cdots 0}_{M}]^T\}, \tag{22}$$

and the existence of a solution for the linear system (18) follows the same equality of (21) immediately.

3.3 Fractional step method

In this subsection, we follow the idea of fractional step method developed by Taira and Colonius [21] to solve the linear system of equations (18). In [21], the authors applied the immersed boundary method to simulate the incompressible flow over solid bodies with prescribed body surface motion. Unlike the previous approaches in [4, 17, 11], they introduced the boundary force as another Lagrange multiplier to enforce the no-slip constraint for the velocity at the immersed boundary. From this point of view, the present approach shares the similar spirit as in [21] by introducing the elastic tension as a new Lagrange multiplier to enforce the surface divergence free constraint (12) along the interface. Since the pressure can be regarded as a Lagrange multiplier for the fluid divergence free constraint (11), one can group those two Lagrange multipliers as a new column vector $\phi = [p, \sigma]^T$ and combine the sub-matrices G and S as $Q = [G, S]$, and the linear system (18) now becomes

$$\begin{bmatrix} L & Q \\ Q^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \phi \end{bmatrix} = \begin{bmatrix} bc_1 \\ \widetilde{b} \end{bmatrix}, \text{where} \, \widetilde{b} = \begin{bmatrix} bc_2 \\ bc_3 \end{bmatrix}. \tag{23}$$

As in [21], we perform a block **LU** decomposition on the above equation to obtain

$$\begin{bmatrix} L & 0 \\ Q^T & -Q^T L^{-1} Q \end{bmatrix} \begin{bmatrix} I & L^{-1}Q \\ 0 & I \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \phi \end{bmatrix} = \begin{bmatrix} bc_1 \\ \widetilde{b} \end{bmatrix}. \tag{24}$$

Note that, the above decomposition is possible (L^{-1} exists), since the matrix L arising from the discrete Laplacian operator is symmetric and negative definite. This matrix decomposition is also referred to as Uzawa method. One can further split the above matrix equation into the following steps by introducing an intermediate velocity vector $\mathbf{u}^*$

$$L\mathbf{u}^* = bc_1, \tag{25}$$

$$(-Q^T L^{-1} Q)\phi = \widetilde{b} - Q^T \mathbf{u}^*, \tag{26}$$

$$\mathbf{u} = \mathbf{u}^* - L^{-1} Q\phi. \tag{27}$$

Recall that the original matrix denoted by A in Eq. (18) is singular due to the pressure value is unique up to a constant, thus the singularity cannot be removed from applying the block LU decomposition. In fact, the matrix $(-Q^T L^{-1} Q)$ in Eq. (26) is symmetric and singular with rank $mn + M - 1$ since G is singular. We now provide an existence of solution for Eq. (26) as follows.

For any vector $y \in \ker((-Q^T L^{-1} Q)^T) = \ker(-Q^T L^{-1} Q)$, then the vector y satisfies $y^T(-Q^T L^{-1} Q)y = (Qy)^T(-L^{-1})(Qy) = 0$. This implies that $Qy = 0$ since $-L^{-1}$ is a positive definite matrix. Thus, $y \in \ker(Q)$. By writing $L \cdot 0 + Qy = 0$, we can immediately obtain the vector $[\,0 \;\; y\,]^T \in \ker(A) = \ker(A^T)$. Notice that, such y also satisfies $y^T(\widetilde{b} - Q^T \mathbf{u}^*) = 0$ since $y^T \widetilde{b} = 0$ due to the discrete incompressible constraint (11). Therefore, the right-hand side vector of Eq. (26) belongs to the range of the matrix $(-Q^T L^{-1} Q)$ in which a solution exists. One can also immediately see from the structure of sub-matrix G that the solution in Eq. (26) is unique up to a constant.

Now we are ready to describe the detailed numerical implementation for solving Eqs. (25)-(27). It is a common practice to avoid the direct computation of the inverse of the matrix L since it is too expensive. In [21], a second-order approximation for L^{-1} based on Taylor expansion is implemented for solving similar equations as our Eqs. (26)-(27), and the conjugate gradient method is applied to solve those equations iteratively. However, this leads to another time step constraint related to the viscosity and the eigenvalues of the discrete Laplacian. In this paper, since we are working on the Stokes flow rather than the Navier-Stokes, we are unable to approximate L^{-1} using Taylor's expansion. Although we do not approximate the L^{-1} directly, we still can solve Eqs. (26)-(27) efficiently thanks to the fast Poisson solver developed in public software package FISHPACK [1]. (The present matrix L is nothing but the discrete Laplacian operator.) The detailed steps for solving Eqs. (25)-(27) are as follows.

Step 1. Solve Eq. (25) by two fast Poisson solvers to obtain intermediate velocity field $\mathbf{u}^*$.

Step 2. Since the matrix $(-Q^T L^{-1} Q)$ is symmetric and positive semi-definite, the conjugate gradient method can be applied. In each iteration, a matrix-vector product $(-Q^T L^{-1} Q)\,\varphi$ is needed. Fortunately, this can be done by letting $z = L^{-1} Q\varphi$, and solving $Lz = Q\varphi$. Once it is done, we multiply z by $-Q^T$ to obtain the product needed. Again, the time consuming cost in each iteration is one fast Poisson solver.

Step 3. Find the velocity field $\mathbf{u}$ from Eq. (27). Since ϕ is solved via Step 2, by solving $Lw = Q\phi$, we then obtain $\mathbf{u} = \mathbf{u}^* - w$. Again, this involves applying one fast Poisson solver.

Therefore, the overall cost in Step 1-3 for our present numerical algorithm can be counted in terms of the number of fast Poisson solver applied. In next section, we shall show the numbers of fast Poisson solver used in the Stokes flow for different grid resolutions.

4 Numerical results

In this section, we perform a series of numerical tests for the present scheme. We first provide the convergence and efficiency tests for the Stokes flow without interfaces. We then perform the convergence test for the Stokes flow with an inextensible interface. Finally, we simulate single interface problems in a shear flow to mimic the vesicle problems in [6] without the bending effect. As expected [24], one can still see the tank-treading motion along the interface. Throughout this section, the computational domain is chosen as $\Omega = [-1, 1] \times [-1, 1]$. All numerical runs were carried out on a PC with 4G RAM using double precision arithmetics.

4.1 Convergence test for Stokes solver

In this subsection, we perform the convergence test and evaluate the efficiency for the present Stokes solver without an interface. The numerical algorithm for solving this problem is exactly same as Step 1-3 described in previous section but with a simpler version; that is, $Q = G$ and $\phi = p$. The different efficient Stokes solver can be found in [20, 2, 16] and the references therein.

Here, we use the following analytic solution so that we can easily compute the errors between the exact and the numerical solutions.

$$u_e(x, y) = \sin x \cos y, \quad v_e(x, y) = -\cos x \sin y \quad \text{and} \quad p_e(x, y) = e^x \sin y.$$

Note that, the above solution does not satisfy the pure Stokes equations so we need to add some external force field (which can be easily computed) into the equations. However, it does not change the method or algorithm since the extra force term appears in the right-hand side of equations. Along the boundary of the computational domain, the Dirichlet boundary conditions for the velocity are provided while no pressure boundary condition is needed in our setting.

It is also worth mentioning that the pressure is unique up to a constant in Stokes equations. Rather than pinning a certain value to a particular discrete pressure as in [21], the uniqueness can be guaranteed by setting up a constraint for the discrete pressure as

$$\sum_{i,j} p_{i,j}\, h^2 = \int_\Omega p_e(\mathbf{x})\, d\mathbf{x}. \tag{28}$$

So our initial guess p_{ij}^0 in conjugate gradient iteration can be chosen as $p_{ij}^0 = \int_\Omega p_e(\mathbf{x})\, d\mathbf{x}/|\Omega|$. By using the mathematical induction, one can easily show that during the conjugate gradient iteration, the discrete pressure does satisfy the constraint (28). In those tests, the tolerance of residual is chosen as 10^{-8}.

Table 1 shows the maximum errors between the exact and numerical solutions for different grid resolutions. One can see that the velocity field has clean second order accuracy while

$m = n$	32	64	128	256	512
$\|u_e - u_h\|_\infty$	1.578e-4	4.481e-5	1.206e-5	3.153e-6	8.120e-7
rate	-	1.82	1.89	1.94	1.96
$\|v_e - v_h\|_\infty$	1.578e-4	4.481e-5	1.206e-5	3.153e-6	8.120e-7
rate	-	1.82	1.89	1.94	1.96
$\|p_e - p_h\|_\infty$	9.615e-4	4.286e-4	2.052e-4	1.005e-4	4.970e-5
rate	-	1.17	1.06	1.03	1.02

Table 1: Numerical accuracy of Stokes solver.

$m = n$	32	64	128	256	512
iterations	12	14	15	16	18
CPU time(sec)	0.02	0.05	0.20	0.93	4.97

Table 2: The cost of CPU time and iterations.

the pressure has clean first-order accuracy. Table 2 shows the efficiency of present Stokes solver. One can see that the number of conjugate gradient iterations increase slightly even when we double the grid sizes and the CPU time for 512×512 mesh is just a few seconds.

4.2 Convergence test for Stokes flow with an inextensible interface

In this subsection, we perform the convergence study for the present numerical algorithm to the Stokes flow with an inextensible interface. Here, we put an inextensible interface Γ with initial configuration $(X(s), Y(s)) = (0.2\cos(s), 0.5\sin(s))$ under a shear flow $(u, v) = (\gamma\, y, 0)$ in a fluid domain Ω, see Figure 1 in detail. The shear rate γ is chosen to be $\gamma = 1$ and the fluid viscosity is $\mu = 1$. We choose the different mesh sizes as $m = n = 64, 128, 256, 512$ so the corresponding mesh is $h = 2/m$. We also set the Lagrangian mesh width as $\Delta s \approx h/2$ and the time step duration for interface evolution as $\Delta t = h/4$.

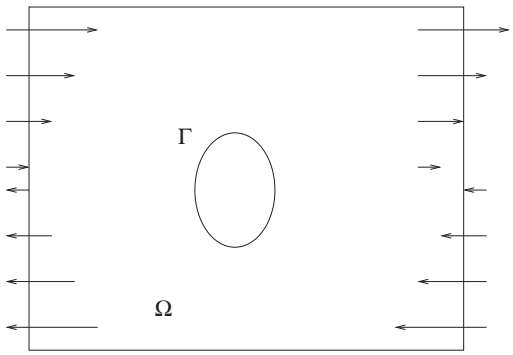

Figure 1: A diagram of an inextensible interface in a shear flow.

	$m = n = 64$	$m = n = 128$	rate	$m = n = 256$	rate
$\lvert L_h - L_0\rvert / L_0$	1.349e-03	7.201e-04	0.91	3.364e-04	1.10
$\lvert A_h - A_0\rvert / A_0$	9.069e-04	4.132e-04	1.14	2.010e-04	1.04
$\lVert \mathbf{X}_h - \mathbf{X}_{\text{ref}}\rVert_\infty$	6.878e-03	2.498e-03	1.46	7.808e-04	1.68
$\lVert u_h - u_{\text{ref}}\rVert_\infty$	4.110e-02	1.675e-02	1.30	5.589e-03	1.46
$\lVert v_h - v_{\text{ref}}\rVert_\infty$	4.050e-02	1.672e-02	1.28	6.086e-03	1.58

Table 3: The mesh refinement results for the perimeter of the interface L_h, the enclosed area A_h, the interface configuration $\mathbf{X}_h$, and the velocity u_h and v_h.

Since the analytical solution is not available in this test, we choose the result obtained from the finest mesh $m = n = 512$ as our reference solution, and compute the maximum error between the reference solution and the numerical solution. All the numerical solutions are computed up to time $T = 0.5$. Since the interface is inextensible and the fluid is incompressible, the perimeter of the interface and the enclosed area by the interface should remain constants theoretically as time goes on. Let L_0 and L_h be the perimeters of the interface at the initial time and final time $T = 0.5$, respectively. The relative error of the perimeter is defined as $|L_h - L_0|/L_0$, and the relative error of the area enclosed is $|A_h - A_0|/A_0$. Table 3 shows the relative errors of the perimeter, the area of the region bounded by the interface, the maximum error of the interface configuration, and the maximum error for the fluid velocity field. Note that, the fluid variables are defined at the staggered grid so when we refine the mesh, the numerical solutions will not coincide with the same grid locations. In these runs, we simply use a linear interpolation to compute the solutions at the desired locations. Due to the fact that the immersed boundary formulation has the singular forcing term in the equations, regularizing the singular term by smoothing discrete delta function causes the method to be first-order accurate. The numerical results shown in Table 3 are consistent with what we expect from theory.

4.3 Tank-treading motion for an inextensible interface under shear flow

Unlike the previous subsection that we focus on the numerical convergence test for our present scheme, in this subsection, we consider the physical transient deformation of an inextensible interface subject to a simple shear flow. As mentioned before, the motivation of this test is to mimic the simulation of the vesicle dynamics which has a lot of applications in bio-fluid problems.

As in the previous test, we put an inextensible interface Γ with initial configuration $(X(s), Y(s)) = (0.18\cos(s), 0.5\sin(s))$ under a shear flow $(u, v) = (\gamma\, y, 0)$ in a fluid domain $\Omega = [-1, 1] \times [-1, 1]$, see Figure 1 in detail. The shear rate γ is chosen to be $\gamma = 1$ and the fluid viscosity is $\mu = 1$. The mesh used is 128×128 and the residual tolerance for is 10^{-4}. It is worth mentioning that the elastic tension σ computed in previous numerical experiments [24, 6, 10] tends to oscillate along the interface which makes the conjugate gradient method for solving in Eq. (26) difficult to converge if the residual tolerance is too large. Figure 2 shows the interface configuration at three different times $t = 0.0625, 1.25, 3.125$, while Figure 3

shows the corresponding elastic tension σ plotted counterclockwise along the interface. (The starting point of zero length in Figure 3 is marked by $\times$ in Figure 2.) One can indeed see some slight oscillations of the tension along the interface as seen in previous literature [24, 6, 10]. Furthermore, one can see that the tension has the smallest values at the interface positions where the curvatures are largest (both tips) which is also in agreement with those in previous literature.

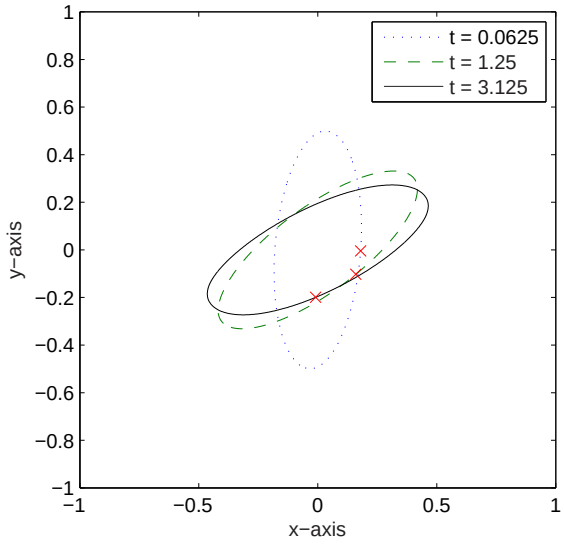

Figure 2: The motion of an inextensible interface in a shear flow with initial configuration $(X(s), Y(s)) = (0.18\cos(s), 0.5\sin(s))$.

To be more physically realistic, we now run the problem for a longer time. It is well-known that the equilibrium dynamics of inextensible interface or vesicle under a simple shear flow undergoes a tanking-treading motion if the viscosity contrast under a certain threshold [7]. Here, by tank-treading motion we mean that the configuration of the interface or vesicle remains stationary while there is a tangential motion along the interface. Figure 4 shows the evolutional motion of the interface (bi-concave shape initially) at different times. We can see that after some time, the interface shape does not seem to change at all. However, the Lagrangian point along the interface marked by "$*$" moves along with its tangential velocity. The streamlines at three different chosen times are shown in Figure 5 in which we can see no normal motion in equilibrium. This tank-treading motion is in a good agreement with previous studies [24, 22, 6, 10].

As discussed in previous literature [24, 8, 9, 22, 6], the motion of a steady or an equilibrium inextensible interface (or vesicle) can be characterized by both the inclination angle θ between the long axis of interface and the flow direction, and the tank-treading frequency $f = 2\pi/\int_\Gamma \frac{dl}{u_\tau}$ of the revolution, where u_τ is the tangential velocity component. The inclination angle has been founded to be strongly dependent on the reduced area $V = \frac{A_0}{\pi R_0^2}$, where A_0 is the enclosed area of the interface, and $R_0 = L_0/(2\pi)$, where L_0 is the total

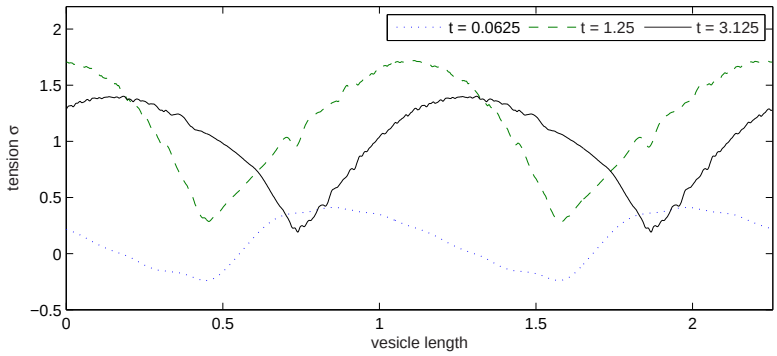

Figure 3: The plots of tension σ along the interface.

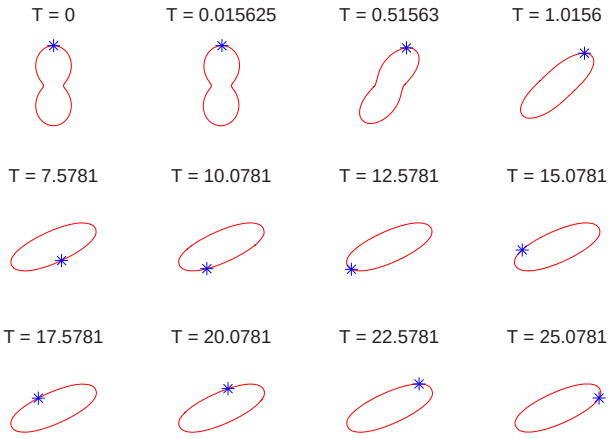

Figure 4: The tank-treading motion of an inextensible interface under a shear flow.

length of interface. (Notice that, by above definition, a circle has the reduced area $V = 1$, while an ellipse with larger aspect ratio has a smaller reduced area.) However, the angle is independent of the shear rate γ. This behavior has been verified in the left panel of Figure 6 and is in good agreement with previous studies [24, 8, 9, 22, 6] which shows the steady inclination angle (θ/π) versus the reduced area (V) with different shear rates $\gamma = 1, 5, 10$. We have observed that the inclination angle increases with the reduced area but is nearly independent of the shear rate. The right panel of Figure 6 shows that the tank-treading frequency f versus the reduced area V for different shear rates. One can see that as the shear rate increases, the tangential motion becomes stronger; thus, the frequency becomes larger. Moreover, by fixing the shear rate, if the interface has larger reduced area then it has larger frequency as well. Again, our numerical results are in a good agreement with previous studies in literature.

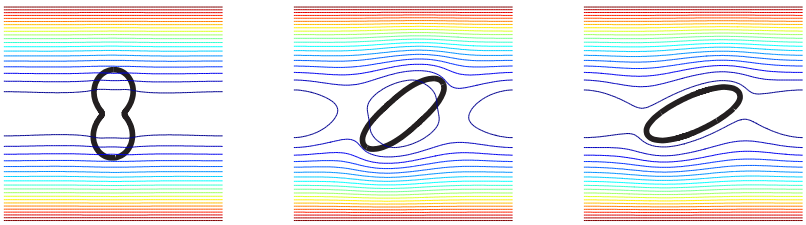

Figure 5: The streamlines of the flow along with an interface.

Appendix

In this appendix, we give a direct derivation to the matrix obtained from the discrete spreading operator S_h of σ_h and the matrix obtained from discrete surface divergence operator ∇_{s_h} of $\mathbf{U}$ are transpose with each other. First, let us rewrite the operator $S_h(\sigma_h)$ by

$$\begin{aligned}
S_h(\sigma_h) &= \sum_{k=0}^{M-1} D_s\left(\sigma\boldsymbol{\tau}\right)_k \delta_h(\mathbf{x}-\mathbf{X}_k)\,\Delta s \\
&= \sum_{k=0}^{M-1} \frac{\sigma_{k+1/2}\,\boldsymbol{\tau}_{k+1/2} - \sigma_{k-1/2}\,\boldsymbol{\tau}_{k-1/2}}{\Delta s}\,\delta_h(\mathbf{x}-\mathbf{X}_k)\Delta s \\
&= \sum_{k=1}^{M} \sigma_{k-1/2}\,\boldsymbol{\tau}_{k-1/2}\,\delta_h(\mathbf{x}-\mathbf{X}_{k-1}) - \sum_{k=1}^{M} \sigma_{k-1/2}\boldsymbol{\tau}_{k-1/2}\delta_h(\mathbf{x}-\mathbf{X}_k) \\
&- \sigma_{-1/2}\boldsymbol{\tau}_{-1/2}\delta_h(\mathbf{x}-\mathbf{X}_0) + \sigma_{M-1/2}\boldsymbol{\tau}_{M-1/2}\delta_h(\mathbf{x}-\mathbf{X}_M) \\
&= \sum_{k=1}^{M} \left(\delta_h(\mathbf{x}-\mathbf{X}_{k-1}) - \delta_h(\mathbf{x}-\mathbf{X}_k)\right)\,\boldsymbol{\tau}_{k-1/2}\,\sigma_{k-1/2}.
\end{aligned}$$

Note that, the last two terms are cancelled out due to the periodicity of the interface. Now we can write down the discrete operator ∇_{s_h} as

$$\begin{aligned}
\nabla_{s_h}\cdot\mathbf{U}_k &= \frac{\mathbf{U}_k - \mathbf{U}_{k-1}}{\Delta s}\cdot\boldsymbol{\tau}_{k-1/2}/\left|D_s\mathbf{X}\right|_{k-1/2} \\
&= -\frac{h^2}{\Delta s\,\left|D_s\mathbf{X}\right|_{k-1/2}} \sum_{\mathbf{x}} \left(\delta_h(\mathbf{x}-\mathbf{X}_{k-1}) - \delta_h(\mathbf{x}-\mathbf{X}_k)\right)\boldsymbol{\tau}_{k-1/2}\cdot\mathbf{u}_{i,j}.
\end{aligned}$$

Since the discrete surface divergence operator of the velocity is zero as described in Eq. (12), we can scale out the coefficient $-\frac{h^2}{\Delta s\,|D_s\mathbf{X}|_{k-1/2}}$ so that the resultant matrices obtained from S_h and $\nabla_{s_h}\cdot$ are transpose to each other.

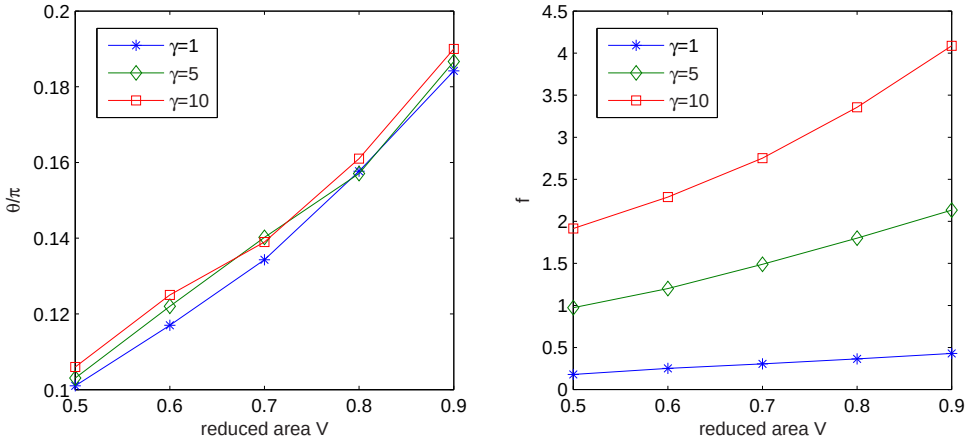

Figure 6: The inclination angles θ/π (left) and the tank-treading frequency $f = 2\pi/\int_\Gamma \frac{\mathrm{d}l}{\mathbf{u}\cdot\boldsymbol{\tau}}$ (right) versus reduced areas V with different shear rates for the tank-treading motions of an inextensible interface in a shear flow.

Acknowledgment

The first author is supported in part by National Science Council of Taiwan under research grant NSC-97-2628-M-009-007-MY3, NSC-98-2115-M-009-014-MY3, and the support of NCTS in Taiwan.

References

[1] J. Adams, P. Swarztrauber, and R. Sweet, Fishpack – a package of Fortran subprograms for the solution of separable elliptic partial differential equations, 1980.

[2] J. H. Bramble, J. E. Pasciak, and A. T. Vassilev, *Analysis of the inexact Uzawa algorithm for saddle point problems*, SIAM J. Numer. Anal., 34 (1997), 1072-1092.

[3] F. H. Harlow and J. E. Welsh, *Numerical calculation of time-dependent viscous incompressible flow of fluid with a free surface*, Phys. Fluids, 8 (1965), 2181-2189.

[4] D. Goldstein, R. Handler and L. Sirovich, *Modeling a no-slip flow boundary with an external force field*, J. Comput. Phys., 105 (1993) 354-366.

[5] K.H. de Haas, C. Blom, D. van den Ende, M.H.G. Duits, and J. Mellema, *Deformation of giant lipid bilayer vesicles in shear flow*, Physical Review E 56, 1997.

[6] Y. Kim and M.-C. Lai, *Simulating the dynamics of inextensible vesicles by the penalty immersed boundary method*, J. Comput. Phys., 229 (2010), 4840-4853.

[7] S.R. Keller and R. Skalak, *Motion of a tank-treading ellipsoldal particle in a shear flow*, Journal of Fluid Mechanics, 120:27-47, 1982.

[8] V. Kantsler and V. Steinberg, *Orientation and dynamics of a vesicle in tank-treading motion in shear flow*, Physical Review Letters 95, 2005.

[9] M. Kraus, W. Wintz, U. Seifert, and R. Lipowsky, *Fluid vesicles in shear flow*, Physical Review Letters 77, 1996.

[10] Z. Li and M.-C. Lai, *New finite difference methods based on IIM for inextensible interfaces in incompressible flows*, East Asian Journal of Applied Mathematics, in press, 2011.

[11] M.-C. Lai and C. S. Peskin, *An immersed boundary method with formal second order accuracy and reduced numerical viscosity*, J. Comput. Phys. 160 (2000) 705-719.

[12] M.-C. Lai, Y.-H. Tseng, and H. Huang, *An immersed boundary method for interfacial flow with insoluble surfactant*, J. Comput. Phys. 227 (2008) 7279-7293.

[13] H. Noguchi and G. Gompper, *Shape transitions of fluid vesicles and red blood cells in capillary flows*, Proceedings of The National Academy of Sciences, vol. 102, 2005.

[14] J.B. Perot, *An analysis of the fractional method*, J. Comput. Phys., 108 (1993), 51-58.

[15] C. Pozrikidis, *The axisymmetric deformation of a red blood cell in uniaxial straining Stokes flow*, Journal of Fluid Mechanics, 216 (1990), 231–254.

[16] J. Peters, V. Reichelt, and A. Reusken, *Fast iterative solvers for discrete Stokes equations*, SIAM J. Sci. Comput., 27 (2005), 646–666.

[17] E. M. Saiki and S. Biringen, *Numerical simulation of a cylinder in uniform flow: application of a virtual boundary method*, J. Comput. Phys., 123 (1996) 450.

[18] M.J. Stevens, *Coarse-grained simulations of lipid bilayers*, Journal of Chemical Physics, 121, 2004.

[19] J.S. Sohn, Y.-H. Tseng, S. Li, A. Voigt and J.S. Lowengrub, *Dynamics of multicomponent vesicles in a viscous fluid*, J. Comput. Phys., 229 (2010), 119-144.

[20] J. C. Strikwerda, *An iterative method for solving finite difference approximations to the Stokes equations*, SIAM J. Numer. Anal., 21 (1984), 447-458.

[21] K. Taira and T. Colonius, *The immersed boundary mehtod: A projection approach*, J. Comput. Phys., 225 (2007), 2118-2137.

[22] 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), 2334–2353.

[23] X. Yang, X. Zhang, Z. Li, and G.-W. He, *A smoothing technique for discrete delta functions with application to immersed bounary method in moving boundary simulations*, J. Comput. Phys., 228 (2009), 7821-7836.

[24] H. Zhou and C. Pozrikidis, *Deformation of liquid capsules with incompressible interfaces in simple shear flow*, Journal of Fluid Mechanics 283 (1995), 175–200.