Commun. Comput. Phys.
doi: 10.4208/cicp.090211.220811s

Vol. **12**, No. 2, pp. 528-543
August 2012

An Iterative Two-Fluid Pressure Solver Based on the Immersed Interface Method

Sheng Xu*

Department of Mathematics, Southern Methodist University, Dallas, TX 75275-0156, USA.

Received 9 February 2011; Accepted (in revised version) 22 August 2011

Available online 20 February 2012

Abstract. An iterative solver based on the immersed interface method is proposed to solve the pressure in a two-fluid flow on a Cartesian grid with second-order accuracy in the infinity norm. The iteration is constructed by introducing an unsteady term in the pressure Poisson equation. In each iteration step, a Helmholtz equation is solved on the Cartesian grid using FFT. The combination of the iteration and the immersed interface method enables the solver to handle various jump conditions across two-fluid interfaces. This solver can also be used to solve Poisson equations on irregular domains.

AMS subject classifications: 76T99, 35J05, 65N06

Key words: Poisson solver, the immersed interface method, two-fluid flow, jump conditions.

1 Introduction

The schematics of an incompressible immiscible two-fluid system is shown in Fig. 1. The two-fluid interface is denoted as $\mathcal{S}$, and its Cartesian coordinates are denoted as $\vec{X}$, as shown in Fig. 1. A single set of conservation equations governing the two-fluid flow reads [11]

$$\rho\left(\frac{\partial\vec{u}}{\partial t}+\nabla\cdot(\vec{u}\vec{u})\right)=-\nabla p+\mu\Delta\vec{u}+\int_{\mathcal{S}}\vec{F}\delta(\vec{x}-\vec{X})\mathrm{d}\mathcal{S}+\rho\vec{g}, \tag{1.1}$$

$$\nabla\cdot\vec{u}=0, \tag{1.2}$$

where $\vec{u}$ is the velocity, p is the pressure, t is time, $\vec{x}$ is Cartesian coordinates, $\vec{F}$ is a force representing interfacial effect, $\delta(\vec{x}-\vec{X})$ is a 3D Dirac delta function, and $\vec{g}$ is a finite

*Corresponding author. *Email address:* sxu@smu.edu (S. Xu)

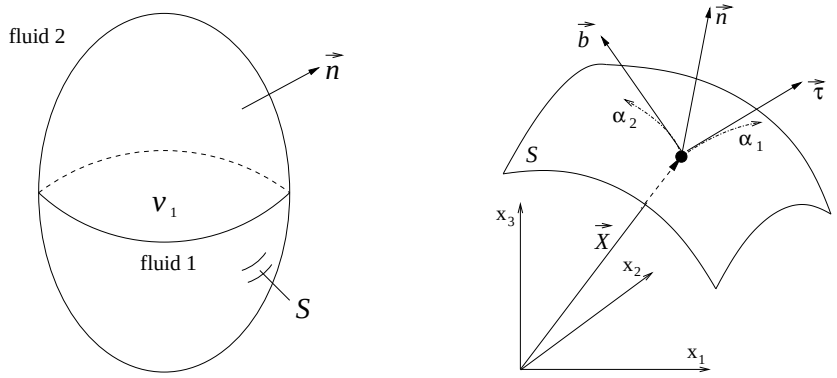

Figure 1: Left: Schematics of a two-fluid system. $\mathcal{V}_1$ is the volume of fluid 1, $\mathcal{S}$ is the two-fluid interface, and $\vec{n}$ is the normal of the interface pointing toward fluid 2; Right: The two-fluid interface $\mathcal{S}$ in a Cartesian coordinate system. $\vec{x}$ is Cartesian coordinates, α_1 and α_2 are two parameters parameterizing the interface locally, $\vec{X}$ is the Cartesian coordinates of the interface, and $\vec{\tau}$, $\vec{b}$ and $\vec{n}$ are unit tangents and normal.

smooth body force. The density ρ and viscosity μ are given by

$$\rho=\rho_1 H(\vec{x},t)+\rho_2(1-H(\vec{x},t)), \tag{1.3}$$
$$\mu=\mu_1 H(\vec{x},t)+\mu_2(1-H(\vec{x},t)), \tag{1.4}$$

where ρ_1 and μ_1 are the constant density and viscosity of fluid 1, ρ_2 and μ_2 are the constant density and viscosity of fluid 2, and $H(\vec{x},t)$ is a 3D step function (Heaviside function) which satisfies

$$H(\vec{x},t)=\begin{cases} 1, & \vec{x}\in\mathcal{V}_1, \\ 0, & \vec{x}\notin\mathcal{V}_1, \end{cases} \tag{1.5}$$

where $\mathcal{V}_1$ is the volume occupied by fluid 1 at time t, as shown in Fig. 1.

By taking the divergence of Eq. (1.1) and applying Eq. (1.2), we obtain a Poisson equation for the pressure p in the two-fluid flow. Away from the two-fluid interface, the Poisson equation reads

$$\Delta p=\rho\nabla\cdot(-\nabla\cdot(\vec{u}\vec{u})+\vec{g}). \tag{1.6}$$

Across the two-fluid interface, the pressure satisfies various jump conditions. In [16], we have derived the following principal jump conditions

$$[p]=\vec{F}\cdot\vec{n}-2[\mu]\left(\frac{\partial\vec{U}}{\partial\tau}\cdot\vec{\tau}+\frac{\partial\vec{U}}{\partial b}\cdot\vec{b}\right), \tag{1.7}$$

$$\left[\frac{1}{\rho}\frac{\partial p}{\partial n}\right]=\frac{\partial}{\partial\tau}\left(\left[\frac{\mu}{\rho}\right]\frac{\partial\vec{U}}{\partial\tau}\cdot\vec{n}-\vec{\tau}\cdot\left[\frac{\mu}{\rho}\frac{\partial\vec{u}}{\partial n}\right]\right)+\frac{\partial}{\partial b}\left(\left[\frac{\mu}{\rho}\right]\frac{\partial\vec{U}}{\partial b}\cdot\vec{n}-\vec{b}\cdot\left[\frac{\mu}{\rho}\frac{\partial\vec{u}}{\partial n}\right]\right), \tag{1.8}$$

$$[\Delta p]=[f], \tag{1.9}$$

where $[\cdot]=(\cdot)_{\vec{n}^+}-(\cdot)_{\vec{n}^-}$ denotes a jump across the interface, $\vec{U}$ is the interface velocity, $\vec{\tau}$, $\vec{b}$ and $\vec{n}$ are unit tangents and normal of the interface as shown in Fig. 1, and f is the right-hand side of Eq. (1.6).

In this paper, we propose an iterative solver based on the immersed interface method to solve for the pressure such that it satisfies the Poisson equation, Eq. (1.6), and the jump conditions, Eqs. (1.7), (1.8) and (1.9).

The immersed interface method [6] is well suited for solving an interface problem with given interfacial jump conditions [4,5]. The key idea of the method is incorporation of necessary jump conditions into a numerical scheme. Take a central finite difference scheme for the second derivative of the function $g(s)$ as an example. Let s_{k-1}, s_k and s_{k+1} be the points of its three-point stencil with spacing h. Suppose the function is discontinuous at $s=\xi$ on the stencil and has the jumps $[g^{(n)}(\xi)]=g^{(n)}(\xi^+)-g^{(n)}(\xi^-)$, $n=0,1,2,\cdots$. Then these jump conditions can be incorporated into the scheme as [6, 12, 13]

$$\frac{\mathrm{d}^2g(s_k^-)}{\mathrm{d}s^2}=\frac{g(s_{k+1}^-)-2g(s_k)+g(s_{k-1}^+)}{h^2}+\mathcal{O}(h^2)+\frac{1}{h^2}\left(\sum_{n=0}^{3}\frac{[g^{(n)}(\xi)]}{n!}(s_{k-1}-\xi)^n\right),\quad s_{k-1}\le\xi\le s_k, \tag{1.10}$$

$$\frac{\mathrm{d}^2g(s_k^-)}{\mathrm{d}s^2}=\frac{g(s_{k+1}^-)-2g(s_k)+g(s_{k-1}^+)}{h^2}+\mathcal{O}(h^2)+\frac{1}{h^2}\left(\sum_{n=0}^{3}\frac{[g^{(n)}(\xi)]}{n!}(s_{k+1}-\xi)^n\right),\quad s_k\le\xi\le s_{k+1}. \tag{1.11}$$

In the current paper, we focus on the two-fluid pressure in 2D. The extension to 3D is straightforward. As shown in Fig. 2, we use the immersed interface method to solve the pressure Poisson equation, Eq. (1.6), on a rectangular domain $\Omega=\Omega^+\cup\Omega^-$ with a two-fluid interface Γ separating Ω^+ and Ω^-. The finite difference scheme given by Eq. (1.10) or (1.11) is applied to discretize the Laplacian operator on a Cartesian grid $(i\delta x, j\delta y)$, which results in the five-point approximation

$$(\Delta p)_{ij}\approx\frac{p_{i-1,j}-2p_{ij}+p_{i+1,j}}{\delta x^2}+\frac{p_{i,j-1}-2p_{ij}+p_{i,j+1}}{\delta y^2}+c_{ij}, \tag{1.12}$$

where p_{ij} is the pressure at the grid point (i,j), and c_{ij} is the jump contribution due to the incorporation of necessary jump conditions. For a five-point stencil that is not cut by the interface, such as the right stencil shown in Fig. 2, c_{ij} is zero and the approximation is of second-order accuracy. Clearly, c_{ij} is nonzero only for a stencil that is cut by the interface, such as the left stencil shown in Fig. 2. Denote the matrix for the discrete Laplacian as L, the vector for the discrete pressure as $\underline{p}$, the vector for the jump contribution as $\underline{c}$, and the vector formed by the discrete right hand side of Eq. (1.6) as $\underline{f}$. The linear system from the finite difference approximation of the Poisson equation, Eq. (1.6), is

$$L\underline{p}+\underline{c}=\underline{f}. \tag{1.13}$$

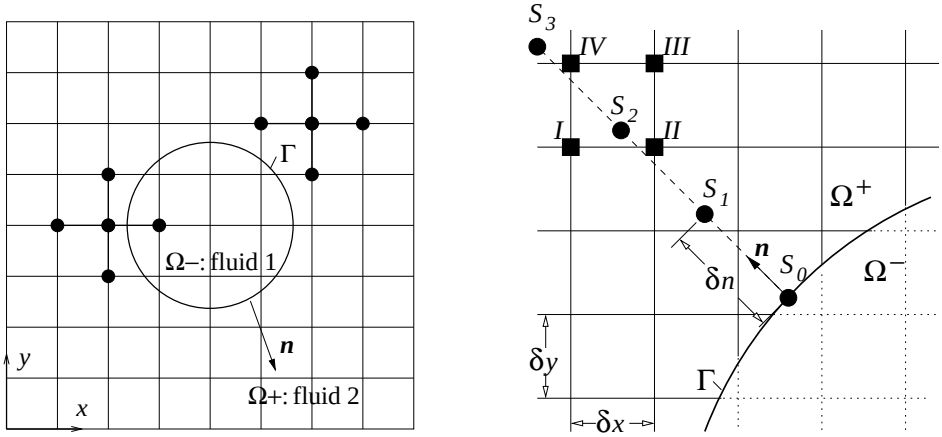

Figure 2: Left: A two-fluid domain discretized by a Cartesian grid; Right: A stencil for one-sided approximation.

Note that the above and later matrix-vector forms are used to facilitate the description of iterative solvers, and we never explicitly form matrices and vectors here and hereafter.

By Eqs. (1.10) and (1.11), it is obvious that the following jump conditions are required to achieve first-order accuracy in the above five-point approximation at grid points near the interface

$$[p], \quad \left[\frac{\partial p}{\partial x}\right], \quad \left[\frac{\partial p}{\partial y}\right], \quad \left[\frac{\partial^2 p}{\partial x^2}\right], \quad \left[\frac{\partial^2 p}{\partial y^2}\right]. \tag{1.14}$$

With second-order accuracy elsewhere, the pressure can then be solved with second-order accuracy in the infinity norm [2]. If the principal jump conditions

$$[p], \quad \left[\frac{\partial p}{\partial n}\right], \quad [\Delta p] \tag{1.15}$$

are explicitly available, the required Cartesian jump conditions in (1.14) can be systematically derived [12, 14], and the jump contribution $\underline{c}$ in Eq. (1.13) is explicitly known and independent of $\underline{p}$. The linear system given by Eq. (1.13) becomes

$$L\underline{p} = \underline{f} - \underline{c}, \tag{1.16}$$

and can be solved very efficiently by just inverting the standard discrete Laplacian L on the regular Cartesian grid. The effect of the interface is included only at the right-hand side of the linear system.

For the two-fluid pressure, we have the principal jump conditions

$$[p], \quad \left[\frac{1}{\rho}\frac{\partial p}{\partial n}\right], \quad [\Delta p], \tag{1.17}$$

given by Eqs. (1.7), (1.8) and (1.9). Because of the discontinuous density ρ, we don't have the desired jump condition $\left[\frac{\partial p}{\partial \mathbf{n}}\right]$ to derive the required Cartesian jump conditions in (1.14). Instead, we have $\left[\frac{1}{\rho}\frac{\partial p}{\partial \mathbf{n}}\right]$ available. More generally, many interface problems have principal jump conditions which can not be used in a simple and straightforward manner to solve the problems on Cartesian grids because they are not in the desired forms as (1.15) to derive necessary Cartesian jump conditions.

The augmented immersed interface method [3, 7] can be used to overcome this difficulty, in which the desirable but unavailable jump conditions are iteratively solved together with the governing equations by GMRES to satisfy the available jump conditions in undesirable forms. In this paper, we propose a different iteration strategy. This strategy has a few advantages. First, it is robust and can handle various forms of jump conditions. Second, unlike GMRES, its use of computer memory is not increasing with iteration. Third, it can be easily parallelized. The last two are especially important for large-scale applications in 3D. We will demonstrate that this strategy can also be employed to solve Poisson equations on irregular domains.

2 Iteration

A desirable but unavailable jump condition can be related to an available jump condition in an undesirable form. For the two-fluid pressure, the desirable but unavailable jump condition is $\left[\frac{\partial p}{\partial \mathbf{n}}\right]$, and the available jump condition in an undesirable form is $\left[\frac{1}{\rho}\frac{\partial p}{\partial \mathbf{n}}\right]$. The relation between them is either one of [16]

$$\left[\frac{\partial p}{\partial \mathbf{n}}\right] = \rho^+\left[\frac{1}{\rho}\frac{\partial p}{\partial \mathbf{n}}\right] + \frac{[\rho]}{\rho^-}\frac{\partial p^-}{\partial \mathbf{n}}, \tag{2.1}$$

$$\left[\frac{\partial p}{\partial \mathbf{n}}\right] = \rho^-\left[\frac{1}{\rho}\frac{\partial p}{\partial \mathbf{n}}\right] + \frac{[\rho]}{\rho^+}\frac{\partial p^+}{\partial \mathbf{n}}. \tag{2.2}$$

To use the relation, an approximation of $\frac{\partial p^-}{\partial \mathbf{n}}$ or $\frac{\partial p^+}{\partial \mathbf{n}}$ is needed. One way to do so, as illustrated in Fig. 2, is to lay down a one-sided stencil in the normal direction, interpolate the stencil points from surrounding Cartesian grid points, and then apply a one-sided finite difference scheme such as [15]

$$\frac{\partial p^+}{\partial \mathbf{n}} = \frac{-5p(S_1)+8p(S_2)-3p(S_3)}{2\delta n} + \mathcal{O}(\delta n^2), \tag{2.3}$$

where $\delta n = (1+\epsilon)\sqrt{\delta x^2+\delta y^2}$ (ϵ is an arbitrary small positive number to keep different stencil points in different grid cells), $p(S_2)$ is interpolated from the Cartesian grid points I, II, III and IV, $p(S_1)$ and $p(S_3)$ are interpolated similarly as $p(S_2)$. However, the desirable but unavailable jump condition $\left[\frac{\partial p}{\partial \mathbf{n}}\right]$ obtained from the above relation and approximation now depends on the unknown pressure p. Consequently, in Eq. (1.13), the jump

contribution $\underline{c}$ is dependent of $\underline{p}$. The dependency is linear and can be written formally as

$$\underline{c}=C\underline{p}+\underline{c}_0, \tag{2.4}$$

where $\underline{c}_0$ is independent of $\underline{p}$. Note that $\underline{c}$, C and $\underline{c}_0$ are sparse with nonzero entries only for grid points near the interface. The linear system, Eq. (1.13), then becomes

$$L\underline{p}+C\underline{p}=\underline{f}-\underline{c}_0. \tag{2.5}$$

Unlike the linear system in Eq. (1.16), the coefficient matrix $L+C$ of the current linear system includes the effect of the interface and is much more difficult to be inverted.

A straightforward approach to solve the current linear system, Eq. (1.16), starts from the formation of the matrix C and the vector $\underline{c}_0$. An iterative linear solver is then used to invert $L+C$. Many methods in the literature [9] share the essence of this approach. However, the formation of the matrix C and the vector $\underline{c}_0$ is a painstaking process with details very much depending on the discretization.

It can be easily noticed that calculating $\underline{c}$ is simple if $\underline{p}$ is given. In other words, if $\underline{p}$ is given, it is simple to calculate the matrix-vector product $C\underline{p}$ without explicitly knowing the matrix C. This fact makes GMRES a tempting solver. However, unlike the augmented immersed interface method, in which the unknowns are defined on the mesh points of interfaces, the unknowns of the current linear system are defined on the whole Cartesian grid points. The memory requirement of GMRES is daunting.

There was an attempt to construct a heuristic iterative solver [8], but such heuristic solvers are not robust as they are sensitive to parameters, has slow convergence, or diverge.

Below, we construct an iteration, which turns out to have reasonably fast convergence and be quite robust. The implementation of the iteration is very simple and needs minimal modification to an existing immersed interface method code which solves Poisson equations with desirable principal jump conditions.

By introducing an unsteady term, the pressure Poisson equation is changed to a parabolic diffusion equation as

$$\frac{\partial p}{\partial t}=\Delta p-f, \tag{2.6}$$

where t is the pseudo time. Its steady-state solution, which is independent of the initial condition, is the solution to the two-fluid pressure. By Eq. (2.5), this diffusion equation can be discretized in space as

$$\frac{\partial \underline{p}}{\partial t}=L\underline{p}+(C\underline{p}+\underline{c}_0)-\underline{f}. \tag{2.7}$$

If we only treat the term $L\underline{p}$ in Eq. (2.7) implicitly by the θ method, we obtain the following iteration

$$\gamma(\underline{p}^{n+1}-\underline{p}^n)=\theta L\underline{p}^{n+1}+(1-\theta)L\underline{p}^n+(C\underline{p}^n+\underline{c}_0)-\underline{f}, \tag{2.8}$$

where γ is the reciprocal of the pseudo-time step Δt, and $\underline{p}^n$ is the grid pressure at the pseudo time $n\Delta t$. This is the iteration we propose in this paper. It can be re-arranged as

$$-(\theta L-\gamma I)\underline{p}^{n+1}=((1-\theta)L+C+\gamma I)\underline{p}^n+\underline{c}_0-\underline{f}, \tag{2.9}$$

where I is an identity matrix. If $(\theta L-\gamma I)^{-1}$ is invertible, the iteration matrix M is

$$M=-(\theta L-\gamma I)^{-1}((1-\theta)L+C+\gamma I). \tag{2.10}$$

Each iteration then requires solving a Helmholtz equation to invert the matrix $L-(\gamma/\theta)I$, which can be done very efficiently using FFT. It can be shown that

$$\underline{p}^{n+1}-\underline{p}^n=M(\underline{p}^n-\underline{p}^{n-1}), \tag{2.11}$$

$$\underline{p}^{n+1}-\underline{p}=M(\underline{p}^n-\underline{p}), \tag{2.12}$$

where $\underline{p}$ is the steady-state solution, so the error behavior in the iteration process is the same as the behavior of $\underline{p}^{n+1}-\underline{p}^n$.

Let $\underline{e}^n=\underline{p}^{n+1}-\underline{p}^n$, then Eq. (2.8) can be written as

$$L\underline{p}^{n+1}+(C\underline{p}^{n+1}+\underline{c}_0)-\underline{f}=((1-\theta)L+C+\gamma I)\underline{e}^n, \tag{2.13}$$

which gives a stopping criterion for the iteration. Let $\delta^2=\min(\delta x^2,\delta y^2)$ and $\underline{p}_e$ be the exact grid pressure. If Eq. (2.5) is a second-order discretization of the pressure Poisson equation, Eq. (1.6), then $\underline{p}=\underline{p}_e+\mathcal{O}(\delta^2)\underline{1}$, where $\underline{1}$ denotes a vector whose infinity norm equals 1, and Eq. (2.5) can be written as

$$L\underline{p}_e+(C\underline{p}_e+\underline{c}_0)-\underline{f}=(L+C)(\mathcal{O}(\delta^2)\underline{1}). \tag{2.14}$$

The comparison between Eqs. (2.13) and (2.14) suggests that the iteration can be stopped when the infinity norm of $\underline{e}^n$ is of the order $\mathcal{O}(\delta^2)$ or less to achieve the second order accuracy of the final grid solution as long as γ is of order $\mathcal{O}(1)$ or less (which should be, as shown later in numerical examples).

Later numerical examples indicate that the proposed iteration always converges reasonably fast as long as the right one of Eqs. (2.1) and (2.2) is chosen. If $\rho^->\rho^+$, then $|[\rho]/\rho^-|<1$, and Eq. (2.1) is used. If $\rho^+>\rho^-$, then $|[\rho]/\rho^+|<1$, and Eq. (2.2) is used. In either case, the coefficient in front of a one-sided approximation is less than one. Otherwise, the iteration may diverge.

The introduction of an unsteady term with the pseudo time to construct an iteration scheme is an old idea. For example, it was used in the alternating direction implicit (ADI) method by Peaceman and Rachford [10] for solving elliptic PDEs; and it was also used in the artificial compressibility method by Chorin [1] for solving the steady Navier-Stokes equations. In this paper, we apply this idea to the immersed interface method to handle jump conditions in undesirable forms for interface and irregular-domain problems.

3 Examples

In this section, three numerical examples are shown to demonstrate the accuracy, efficiency and robustness of the proposed iteration strategy. The first two examples are interface problems. The third one is an irregular-domain problem.

3.1 1D interface problem

Starting from the following discontinuous exact solution

$$u(x)=\begin{cases}\sin x, & x\in[-1,0.1),\\ e^x, & x\in[0.1,1],\end{cases} \tag{3.1}$$

the following interface problem is constructed

$$\frac{\mathrm{d}^2u}{\mathrm{d}x^2}=\begin{cases}-\sin x, & x\in[-1,0.1),\\ e^x, & x\in[0.1,1],\end{cases}\qquad \left.\frac{\mathrm{d}u}{\mathrm{d}x}\right|_{-1}=\cos(1),\quad \left.\frac{\mathrm{d}u}{\mathrm{d}x}\right|_{1}=e. \tag{3.2}$$

This 1D interface problem is solved on N grid points using the proposed iterative solver with only the following available jump conditions at $x=0.1$

$$[u],\quad \left[\kappa\frac{\mathrm{d}u}{\mathrm{d}x}\right],\quad \left[\frac{\mathrm{d}^2u}{\mathrm{d}x^2}\right], \tag{3.3}$$

where κ is a piecewise positive constant coefficient which is discontinuous at $x=0.1$ with the jump $[\kappa]=\kappa^+-\kappa^-$. The following relation is used to obtain the desired but unavailable jump condition

$$\left[\frac{\mathrm{d}u}{\mathrm{d}x}\right]=\begin{cases}\dfrac{1}{\kappa^+}\left[\kappa\dfrac{\mathrm{d}u}{\mathrm{d}x}\right]-\dfrac{[\kappa]}{\kappa^+}\dfrac{\mathrm{d}u^-}{\mathrm{d}x}, & \kappa^+>\kappa^-\left(\frac{[\kappa]}{\kappa^+}<1\right),\\ \dfrac{1}{\kappa^-}\left[\kappa\dfrac{\mathrm{d}u}{\mathrm{d}x}\right]-\dfrac{[\kappa]}{\kappa^-}\dfrac{\mathrm{d}u^+}{\mathrm{d}x}, & \kappa^+<\kappa^-\left(\frac{[\kappa]}{\kappa^-}<1\right).\end{cases} \tag{3.4}$$

By normalization, it is sufficient to consider the ratio κ^+/κ^-.

Because of the Neumann boundary conditions, the solution of this problem is subject to an arbitrary constant, and one eigenvalue of the iteration matrix M is 1, as shown in Fig. 3. The eigenvalue of M with the second largest amplitude, denoted as r, thus reflects the convergence rate. There are two parameters, θ and γ, in the iterative solver. It is expected that the convergence of the iteration is mainly determined by the implicity parameter θ, and the value of γ, the reciprocal of the pseudo time step, mainly affects the convergence rate. This expectation is confirmed in Fig. 4. Fig. 4 indicates that the iteration is convergent ($r<1$) for $\theta>0.5$ and all values of γ, and the convergence rate r is insensitive to γ if $\gamma<\mathcal{O}(10^{-2})$ and insensitive to N, the number of grid points. The fastest convergence rate (the smallest value of r) is achieved when $\theta\approx1$ and $\gamma<\mathcal{O}(10^{-2})$.

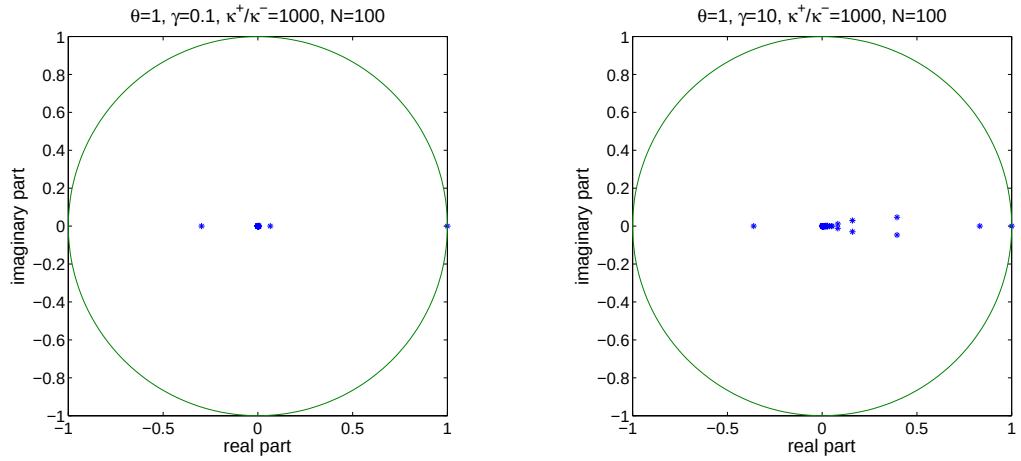

Figure 3: The eigenvalue distribution of the iteration matrix.

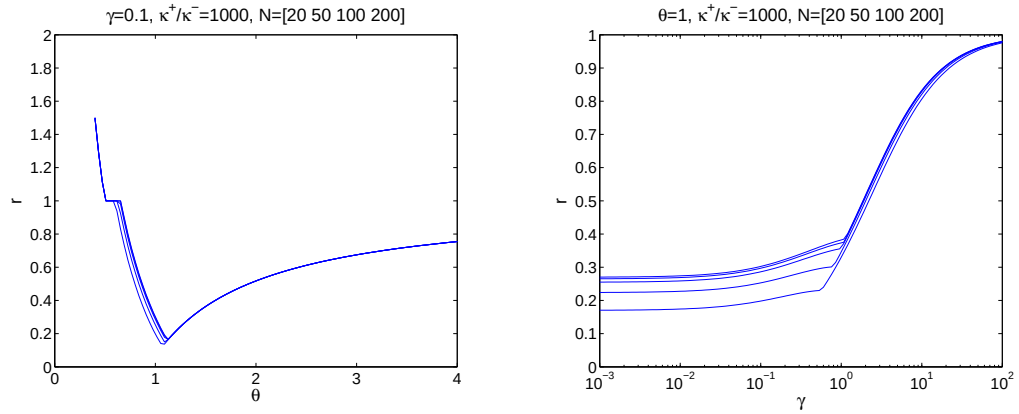

Figure 4: The error reducing factor r as a function of θ and γ.

By Eq.(3.4), the ratio κ^+/κ^- only affects the common factor $|[\kappa]/\kappa^+|$ or $|[\kappa]/\kappa^-|$ of the entries of the matrix C in the iteration matrix M. This common factor approaches 1 and the convergence rates keeps almost unchanged if $\kappa^+/\kappa^- < \mathcal{O}(10^{-2})$ or $\kappa^+/\kappa^- > \mathcal{O}(10^2)$, as demonstrated in Fig. 5: Left. It is therefore expected that the iteration is convergent for all values of the ratio κ^+/κ^-. However, if the conditions in the Eq. (3.4) are switched to make the common factor greater than 1, the iteration only converges for a very small range of the ratio κ^+/κ^- and diverges dramatically for large or small ratio κ^+/κ^-.

That the error behavior in the iteration process is the same as the behavior of $\underline{p}^{n+1}-\underline{p}^n$, where n is the iteration number, is confirmed in Fig. 6: Left. The second-order accuracy of the final solution is clearly indicated by Fig. 6: Right. As analyzed before, the second-order accurate final solution is obtained when the infinity norm of the $\underline{p}^{n+1}-\underline{p}^n$ is of the order $\mathcal{O}(N^{-2})$.

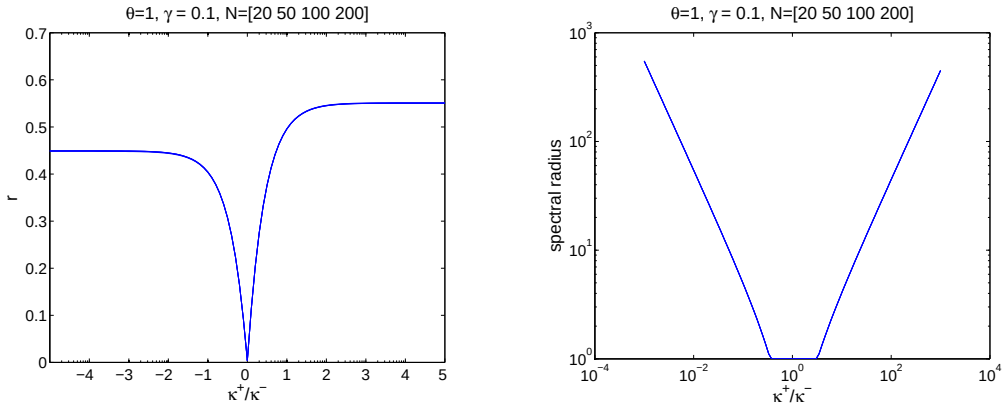

Figure 5: Left: The error reducing factor as a function of κ^+/κ^-; Right: The spectral radius as a function of κ^+/κ^- if the conditions in Eq. (3.4) are switched. Curves for different N are barely distinguishable.

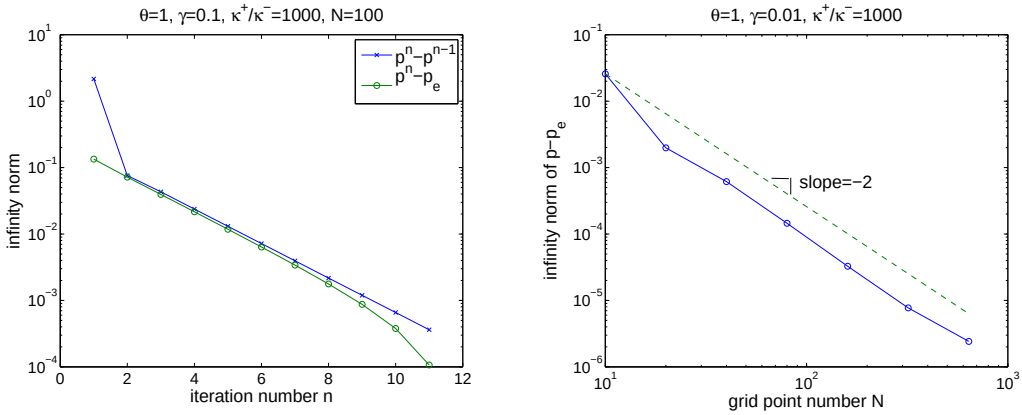

Figure 6: The infinity norm of errors.

3.2 2D interface problem

Starting from the following discontinuous exact solution on a square domain $[-1,1]\times[-1,1]$ with a circular interface $\Gamma=\{(x,y)|x^2+y^2=0.5^2\}$

$$p(x,y)=\begin{cases}\sin x\sin y, & x^2+y^2>0.5^2,\\ e^{-(x+y)}, & x^2+y^2<0.5^2,\end{cases} \tag{3.5}$$

the Poisson equation

$$\Delta p=f \tag{3.6}$$

is solved with analytical Neumann boundary conditions at the far-field boundaries and the following analytical jump conditions at the interface Γ

$$[\kappa p],\quad \left[\frac{1}{\rho}\frac{\partial p}{\partial \mathbf{n}}\right],\quad [\Delta p], \tag{3.7}$$

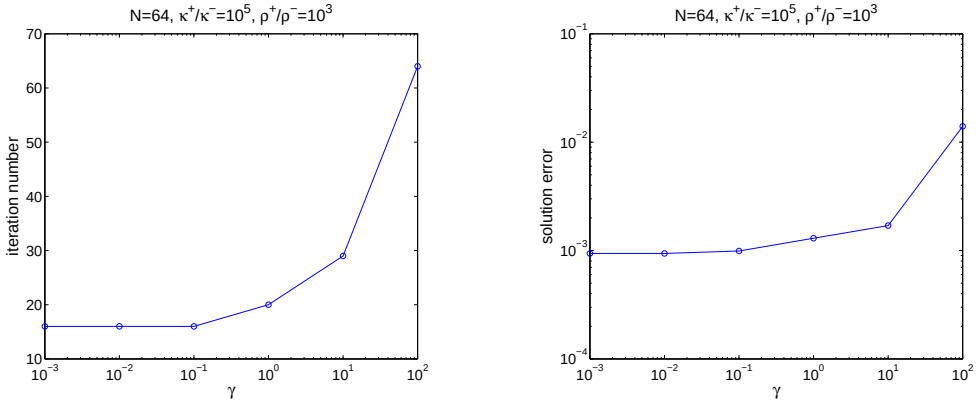

Figure 7: The iteration number and the solution error as functions of γ.

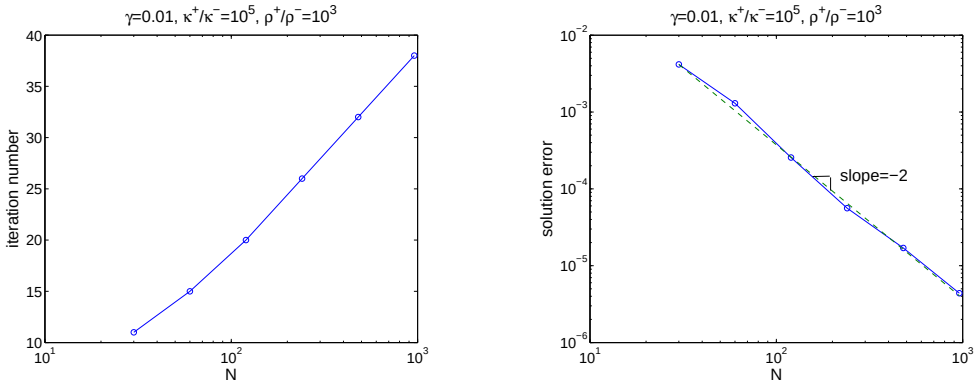

Figure 8: The iteration number and the solution error as functions of N.

where κ and ρ are piecewise positive constants which are discontinuous at the interface Γ with the jump $[\kappa]$ and $[\rho]$. This 2D interface problem is lack of the two desirable jump conditions, $[p]$ and $\left[\frac{\partial p}{\partial \mathbf{n}}\right]$. Besides the relation given by Eq. (2.1) or (2.2), the following relation is also used

$$[p]=\begin{cases} \dfrac{1}{\kappa^+}[\kappa p]-\dfrac{[\kappa]}{\kappa^+}p^-, & \kappa^+>\kappa^-\left(\dfrac{[\kappa]}{\kappa^+}<1\right), \\ \dfrac{1}{\kappa^-}[\kappa p]-\dfrac{[\kappa]}{\kappa^-}p^+, & \kappa^+<\kappa^-\left(\dfrac{[\kappa]}{\kappa^-}<1\right). \end{cases} \tag{3.8}$$

where p^- or p^+ is approximated by interpolation. Note that $\kappa^+=\kappa^-=1$ for the two-fluid pressure. This 2D interface problem is solved below on $N\times N$ Cartesian grid points and $2N$ interface marker points using the proposed iterative solver with $\theta=1$.

Similar to the previous 1D problem, the iteration number and the solution error for this 2D problem are insensitive to γ if $\gamma<\mathcal{O}(10^{-2})$, as indicated in Fig. 7. For this reason, $\gamma=0.01$ is used in later simulations. In all simulations, the iteration is stopped when the infinity norm of $\underline{e}^n=\underline{p}^{n+1}-\underline{p}^n$ is $\mathcal{O}(N^{-2})$. Fig. 8 indicates that the iteration number

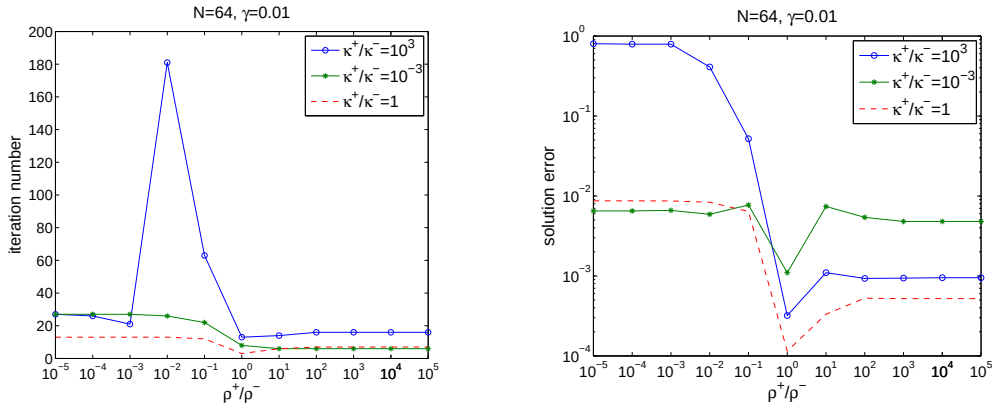

Figure 9: The iteration number and the solution error as functions of ρ^+/ρ^-.

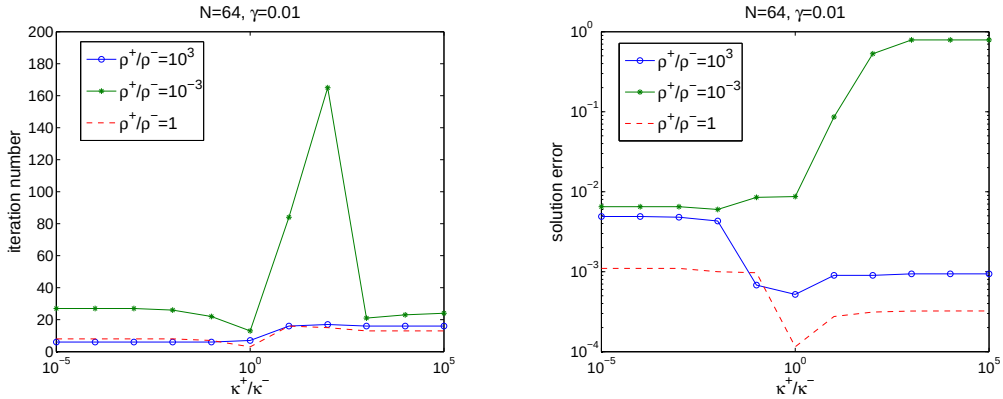

Figure 10: The iteration number and the solution error as functions of κ^+/κ^-.

is increased by almost a constant (about 5) if N is doubled. The convergence rate r is therefore almost independent of N and can be estimated as $r=\left(\frac{1}{2^2}\right)^{1/5}=0.76$. It takes 20 iteration steps or so to bring the error from $\mathcal{O}(1)$ to $\mathcal{O}(10^{-3})$. The grid refinement study in Fig. 8 also indicates the final solution is of second-order accuracy in the infinity norm.

To demonstrate the robustness of the proposed iteration strategy, we consider a wide range of the ratios ρ^+/ρ^- and κ^+/κ^-. Fig. 9 and 10 show the results. In summary, the iteration behaves very well for various combinations of the two ratios except for the combinations of very small ρ^+/ρ^- but very large κ^+/κ^- (lines with open circles in Fig. 9 and 10). In the exception, a lot of iteration steps may be required to satisfy the iteration stopping criterion, and the errors in the final solutions are generally unacceptable. The exception requires the approximations of p^- and $\frac{\partial p^-}{\partial \mathbf{n}}$ at the inside (the concave side) of the interface. The coefficients $-[\kappa]/\kappa^+$ and $[\rho]/\rho^-$ in front of the approximations are very close to -1. It turns out (without showing results here) that the big errors occur only inside the interface, and the solutions outside are still almost second-order accurate. One

attempt to remove the exception is to use change of variables. By changing the unknown from p to $q=\kappa p$, this 2D interface problem can be recast as solving the Poisson equation

$$\Delta q=\kappa f, \tag{3.9}$$

with the jump conditions

$$[q], \quad \left[\frac{1}{\kappa\rho}\frac{\partial q}{\partial \mathbf{n}}\right], \quad [\Delta q]. \tag{3.10}$$

Since ρ^+/ρ^- is very small, and κ^+/κ^- is very large, the amplitude of $(\kappa\rho)^+/(\kappa\rho)^-$ is intermediate. The proposed iteration strategy can be used to solve q with fast convergence. However, the final solution p is still inaccurate if κ^+/κ^- is out of the range $\mathcal{O}(0.1)-\mathcal{O}(10)$ (no results shown here). How to successfully handle the exception will be a future research subject.

We now change the interface in this problem to a more complex one and compare our current iteration with GMRES iteration in the augmented immersed interface method [3, 7]. We use the GMRES solver in the Intel math kernel library and set its convergence tolerance to 10^{-8} and residual tolerance to 10^{-3}. In polar coordinates (θ,r), the new interface is given as

$$r(\theta)=0.5(1+0.25\sin(5\theta)), \tag{3.11}$$

as shown in Fig. 11. The convergence results for the new interface problem are given in Fig. 11. The results indicate that both iterations produce overall second-order accurate solutions. To compare the efficiency of the two iterations, we plot the iteration number and the CPU time as a function of the grid resolution in Fig. 12. The iteration number increases with the grid resolution in our iteration as we expect, and the iteration number in GMRES stays almost unchanged, but the CPU time of our iteration is comparable to GMRES at all the grid resolutions for this case.

3.3 Irregular-domain problem

The proposed iteration strategy can also be used to solve irregular-domain problems. For example, a Poisson equation constructed from the exact solution $p(x,y)=x^2+y^2$ can be solved with the analytical Neumann boundary condition at a rounded rectangular boundary Γ. To solve this exterior problem with the immersed interface method on a fixed grid in the rectangular domain $[-1,1]\times[-1,1]$, the solution inside Γ is set to zero. The available jump conditions across Γ are $\left[\frac{\partial p}{\partial \mathbf{n}}\right]$ and $[\Delta p]$, which come from the given Neumann boundary condition on $\Gamma$and the given right hand sides of the Poisson equation. The missing jump condition $[p]=p^+-p^-=p^+$ is approximated with interpolation from the grid values of p outside Γ. The linear system from discretization of the problem is in the same form as Eq. (2.5), which is then solved by the proposed iteration strategy. An interior problem can be constructed and solved similarly. Shown in Fig. 13 are the contour plots of the results. The convergence speed and the accuracy are comparable to the typical results in previous 2D example.

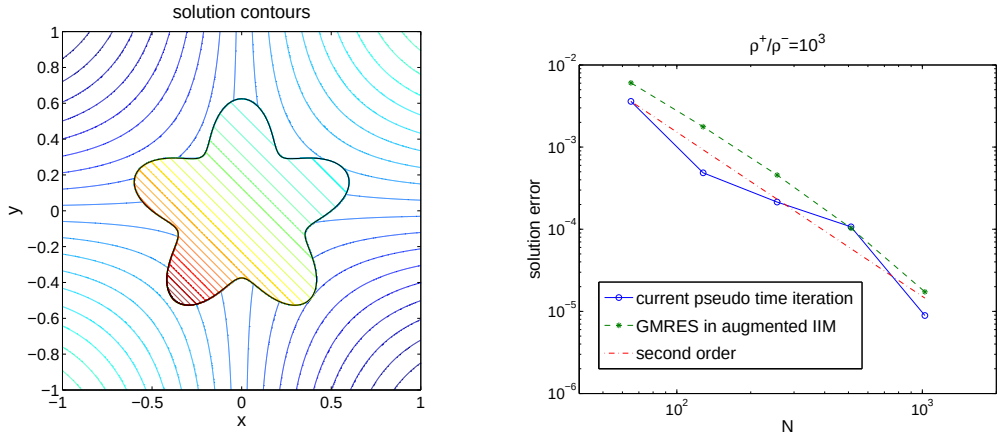

Figure 11: Left: Contours of the solution. Right: The solution error as a function of the grid resolution.

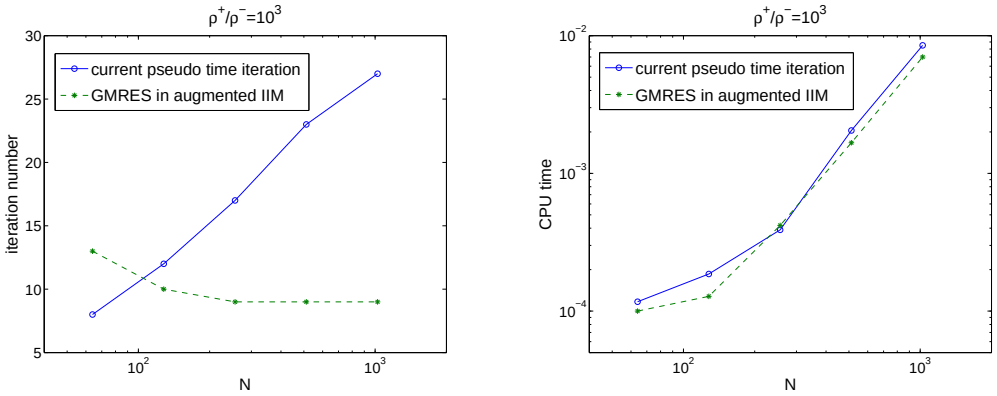

Figure 12: The iteration number and the CPU time as a function of the grid resolution.

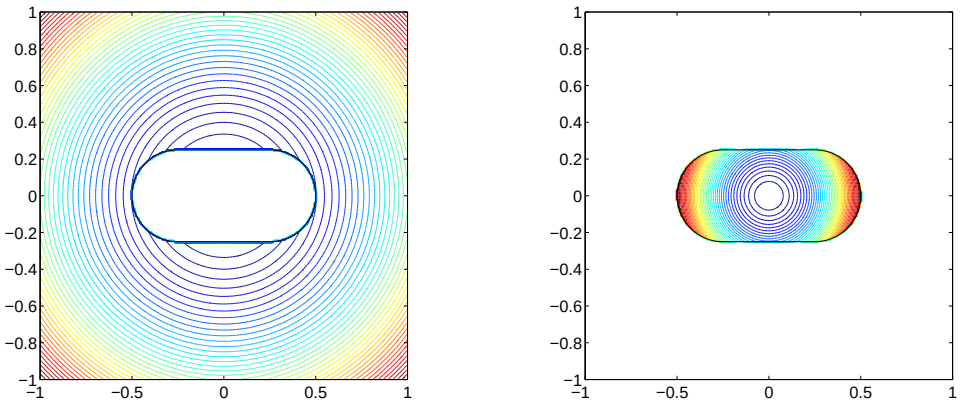

Figure 13: Contour plots of the solutions to an exterior (Left) and interior (Right) irregular-domain problem.

4 Conclusions

An iterative solver based on the immersed interface method is proposed to solve the two-fluid pressure Poisson equation on a fixed Cartesian grid in a rectangular region. The iteration strategy is the introduction of an unsteady term in the Poisson equation and the implicit treatment of the unsteady term. This iterative solver has a reasonably fast convergence speed and allows for FFT solving in each iteration step. The second-order accuracy in the infinity norm can be achieved for the final solution. The solver is robust to handle all density ratios of the two fluids. It can be used to solve Poisson equations subject to various jump conditions on immersed interfaces and Poisson equations on irregular domains.

Acknowledgments

S. Xu thanks the support of this work by the NSF grant DMS 0915237.

References

[1] A. J. Chorin, A numerical method for solving incompressible viscous flow problems, J. Comput. Phys., 2 (1967), 12-26.
[2] H. Huang and Z. Li, Convergence analysis of the immersed interface method, IMA J. Numer. Anal., 19 (1999), 583-608.
[3] K. Ito, M.-C. Lai and Z. Li, A well-conditioned augmented system for solving Navier-Stokes equations in irregular domains, J. Comput. Phys., 228 (2009), 2616-2628.
[4] 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), 1019-1044.
[5] R. J. LeVeque and Z. Li, Immersed interface methods for Stokes flow with elastic boundaries or surface tension, SIAM J. Sci. Comput., 18 (1997), 709-735.
[6] Z. Li and K. Ito, The immersed interface method – numerical solutions of PDEs involving interfaces and irregular domains, SIAM Frontiers in Applied Mathematics, 33 (2006), ISBN: 0-89971-609-8.
[7] Z. Li, X. Wan, K. Ito and S. R. Lubkin, An augmented approach for the pressure boundary condition in a Stokes flow, Commun. Comput. Phys., 1 (2006), 874-885.
[8] 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), 157-192.
[9] R. Mittal and G. Iaccarino, Immersed boundary methods, Annu. Rev. Fluid Mech., 37 (2005), 239-261.
[10] D. Peaceman and H. Rachford, The numerical solution of elliptic and parabolic differential equations, J. SIAM, 3 (1955), 28-41.
[11] G. Tryggvason et al., *A front-tracking method for the computations of multiphase flow*, J. Comput. Phys., 169 (2001), 708-759.
[12] S. Xu and Z. J. Wang, Systematic derivation of jump conditions for the immersed interface method in three-dimensional flow simulation, SIAM J. Sci. Comput., 27 (2006), 1948-1980.

[13] 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), 454-493.
[14] S. Xu and Z. J. Wang, A 3D immersed interface method for fluid-solid interaction, Comput. Methods Appl. Mech. Engrg., 197 (2008), 2068-2086.
[15] S. Xu, The immersed interface method for simulating prescribed motion of rigid objects in an incompressible viscous flow, J. Comput. Phys., 227 (2008), 5045-5071.
[16] S. Xu, Derivation of principal jump conditions for the immersed interface method in two-fluid flow simulation, Discrete and Continuous Dynamical Systems, Supplement 2009 (2009), 838-845.