

國立交通大學

應用數學系

碩 士 論 文

內嵌介面問題的數值方法與應用

Numerical Methods and Applications
for Immersed Interface Problems

研 究 生：曾孝捷

指 導 教 授：賴明治 教授

中 華 民 國 九 十 五 年 六 月

內嵌介面問題的數值方法與應用

Numerical Methods and Applications
for Immersed Interface Problems

研究生：曾孝捷

Student : Hsiao-Chieh Tseng

指導教授：賴明治

Advisor : Ming-Chih Lai

國立交通大學
應用數學系
碩士論文



A Thesis
Submitted to Department of Applied Mathematics
College of Science

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

in

Applied Mathematics

July 2006

Hsinchu, Taiwan, Republic of China

中華民國九十五年六月


內嵌介面問題的數值方法與應用

學生：曾孝捷

指導教授：賴明治 教授

國立交通大學應用數學系（研究所）碩士班

摘 要



本論文介紹計算有不連續條件的二維 Poisson 方程的一個簡單快速的數值方法，並應用本方法來解 Stokes 方程和 Navier-Stokes 方程。在本文中，提供在計算區域上，處理介面的幾何形狀以及在介面上方程的不連續條件。我們使用三次雲線內插來近似不連續條件所在之介面，以及使用二階精度的外插公式來處理不連續條件。在效率的考量上，使用快速的 Poisson 算法。我們給出一些計算數據，來觀察此方法的精度以及應用。

Numerical Methods and Applications for Immersed Interface Problems

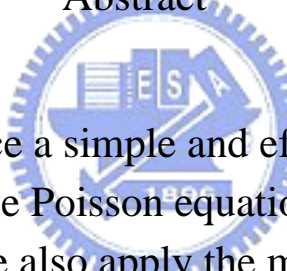
Student: Hsiao-Chieh Tseng

Advisor: Dr. Ming-Chih Lai

Department (Institute) of Applied Mathematics

National Chiao Tung University

Abstract



In this paper, we introduce a simple and efficient second order numerical method to solve Poisson equation with jump conditions in two-dimension (2-D). We also apply the method to simulate Stokes and unsteady Navier-Stokes flow dynamics. One of the essential components of the method is to handle the geometric of the immersed interface and jump discontinuities along it in the computational domain. Cubic spline interpolation is implemented to deal with the interface and a second order accurate extrapolation is applied to incorporate these jump conditions. We also apply the marker-and-cell (MAC) method cooperating with the interface. In the matter of efficiency, we can apply some fast Poisson solvers such as Fishpack. We verify the accuracy, efficiency of our method by showing some numerical experiments.

誌 謝

本篇論文的完成，首先要感謝我的指導老師 賴明治教授。記得剛上研究所，老師帶著懵懂的我，一步一步地踏入科學計算這個領域，指引我學習與研究的方向，並不時給予我鼓勵。從中，不僅學到許多相關的知識與研究的方法，更讓我在這片領域中，找到了興趣與成就感。除了指導老師之外，實驗室的夥伴們也給予我不少助力。曾昱豪學長是我常請教的對象，他豐富的經驗與資源，讓我在研究的過程中，找到明確的方向不至於迷惘；吳恭儉學長以他深厚的數學分析能力，提供我許多在理論方面的驗證與推導。還有李雅羚同學與陳冠羽學弟和我在課業與研究上的切磋，以及許多以畢業學長姐留下來的資料。學生在此謹致最誠摯的謝意。

在論文口試期間，承蒙牛仰堯教授、楊肅煜教授，張書銘教授費心審閱並提供許多寶貴意見，使本論文之完稿得以更加齊備，學生永銘在心。

在研究生活外，我也要感謝我的研究室好夥伴：陳偉國、涂芳婷、楊川和。除了課業上的討論之外，在閒暇之餘和他們話家常，交換各種心得與想法，使我增廣見聞及開拓生活圈，在彼此的相互鼓勵下，一同度過這段歲月。

最後，要感謝我的家人給我的關心與祝福，以及女友蔡琬琦的呵護及支持。這是我心靈的寄託與避風港。他們對我的付出，讓我可以順利地，專心完成我的學業。願與他們，以及所有關心我的人，分享此篇論文完成之喜悅與榮耀。

目 錄

中文提要	i
英文提要	ii
誌謝	iii
目錄	iv
一、	Introduction	1
二、	Poisson Equation with Jump Discontinuities	2
2.1	Introduction	2
2.2	Discontinuous Poisson Problem	2
2.3	Cubic Spline Representation	3
2.3.1	A Brief Introduction to Cubic Spline Representation	4
2.3.2	Solving the Linear System $Ax = b$ with Specific Matrix A	5
2.3.3	Flag Construction	6
2.4	Interpolation from Grid to Interface	7
2.5	Numerical Example	7
三、	Stokes Flow with Singular Force	11
3.1	Stokes Equation	11
3.2	Marker and Cell Method	11
3.3	Numerical Scheme	12
3.4	Handling the Derivatives with a Crossing Interface	13
3.5	Numerical Example	14
四、	Navier-Stokes Equation	18
4.1	Introduction to Navier-Stokes Equation	18
4.2	Numerical Scheme	21
4.3	Numerical Example	23
4.3.1	Accuracy Check without Interfaces	23
4.3.2	Cavity Flow	24
4.3.3	Circular Flow with a Line Force	26
五、	Concluding Remarks	28
Reference	29

1 Introduction

In problems with interfaces, the unknowns or their derivatives may have jump discontinuities. For example, fluid flow or wave propagation an interface between different regions exerts a force on the material, or an interface separates regions of different material properties. The interface may be a fixed obstacle, or be considered as an elastic membrane. The interface is often embedded in a regular computational domain, and varieties of fast methods can apply with some modifications. On the other hand, it also need to be considered that the accuracy, especially near the interface.

There are lots of treatments for the immersed interface problems. Mayo in this paper [9] used the fundamental solutions of Poisson equations to give some integral formulas on the interface. These seems to be some relations between Mayo's treatment and the boundary element method.

In the 1970-80's, C. S. Peskin developed the Immersed Boundary Method (IBM) to simulate the blood flow through heart valves. The valve leaflet (the interface) exerts some force into the blood (the fluid) and t the same time moves along with the local fluid velocity simultaneously. The interaction between the fluid and immersed boundary can be modeled by a well chosen discretized approximation to the Dirac delta function, which is called a discrete delta function. However, it is also known that generally the IBM is only first-order accurate for nonsmooth but continuous quantities. Different variations of IBM have been proposed to improve the accuracy. For example, Lai and Peskin [10] proposed a new formally second-order IBM with reduced numerical viscosity, and Griffith and Peskin [11] provided another higher order method for sufficiently smooth problems.

In contrast to the discrete delta function approach, Leveque and Li proposed Immersed Interface Method (IIM). It translates the jump discontinuities into the local coordinate system of the interface, and then find the correction terms to modify the system in Cartesian grid. They apply this method to solve some fluid problems and obtain second-order accurate [8]. On the other hand, Jane Wang and her group computed the correction terms by extrapolation the jump discontinuities from the interface to the Cartesian grid [12]. They also made some applications in fluid mechanics [13].

The Ghost Fluid Method (GFM), which is also a sharp interface method that uses the jump conditions in the solution and the flux, has been developed by Fedkiw, Liu, Kang and their group for elliptic interface problems. The GFM is simpler than IIM and the resulting linear system of equations for the self-adjoint elliptic equations is symmetric positive definite. The trade-off is that the solution is first-order accurate in the maximum norm.

In this paper, we implement the extrapolation method in [12] and apply it to Poisson, Stokes, and full Navier-Stokes equation.

2 Poisson Equation with Jump Discontinuities

2.1 Introduction

Given a scalar function ϕ with real spatial variable x , we denote

$$[\phi(x_*)] = \phi(x_*^+) - \phi(x_*^-),$$

the right limit subtract the left limit at x_* . We call ϕ has a *jump* at x_* if $[\phi(x_*)] \neq 0$.

Our numerical method is based on *finite difference method* (FDM) with Cartesian Grid points. We denote the *irregular points* whose difference stencil crosses the jump, and the others are called *regular points*.

Taking an 1-D example. Consider the case that one wants to obtain some approximation of derivatives of ϕ at x_j with grid size h but there is a jump of ϕ at x_* between the irregular points x_j and x_{j+1} . Let $x_{j+1} = x_* + \alpha$, Taylor expansion tells us that

$$\begin{aligned} \phi(x_{j+1}) &= \phi(x_*^+ + \alpha) \\ &= \phi(x_*^+) + \phi'(x_*^+)\alpha + \phi''(x_*^+)\frac{\alpha^2}{2} + \mathcal{O}(h^3) \\ &= \phi(x_*^-) + \phi'(x_*^-)\alpha + \phi''(x_*^-)\frac{\alpha^2}{2} + C_{j+1}^{(\phi)} + \mathcal{O}(h^3) \end{aligned}$$

where

$$C_{j+1}^{(\phi)} = [\phi] + [\phi']\alpha + [\phi'']\frac{\alpha^2}{2}$$

is the correction term of $\phi(x_{j+1})$ for x_j . Hence, for example we can obtain

$$\begin{aligned} \phi'(x_j) &= \frac{(\phi(x_{j+1}) - C_{j+1}^{(\phi)}) - \phi(x_{j-1}))}{2h} + \mathcal{O}(h^2), \\ \phi''(x_j) &= \frac{(\phi(x_{j+1}) - C_{j+1}^{(\phi)}) - 2\phi(x_j) + \phi(x_{j-1}))}{h^2} + \mathcal{O}(h). \end{aligned}$$

2.2 Discontinuous Poisson Problem

We consider the interface problem

$$\begin{aligned} \Delta\psi_{\pm} &= \omega_{\pm} \text{ in } \Omega_{\pm}, \\ [\psi] &= q_0 \text{ on } \Gamma, \\ \left[\frac{\partial\psi}{\partial n}\right] &= q_1 \text{ on } \Gamma, \\ \psi &= \psi_b \text{ on } \partial\Omega, \end{aligned} \tag{2.1}$$

where ω_{\pm}, ψ_{\pm} are defined on $\bar{\Omega}_{\pm}$ and q_0, q_1 are on Γ ; with $\Gamma \subseteq \Omega, \bar{\Omega}_- = \Omega_- \cup \Gamma, \bar{\Omega}_+ = \Omega_+ \cup \Gamma$. We assume that ω_{\pm}, q_0, q_1 and Γ are fairly smooth, with a possible jump in ω_{\pm} at Γ .

We generate the 2-D Cartesian grid with grid size $h = \Delta x = \Delta y$, and overlay an irregular boundary Γ with known discontinuities in $[\psi], \left[\frac{\partial\psi}{\partial n}\right]$. Here we assume that Γ can be represented by a periodic cubic spline. In our approach we find the *foot of perpendicular* on Γ for every irregular point. In some sense this is the most suitable location to be chosen

since we know the discontinuities in terms of the normal derivatives. Instead of finding $\left[\frac{\partial^2\psi}{\partial n^2}\right]$, we apply the identity of Laplace operator on a curve, i.e.,

$$\omega = \frac{\partial^2\psi}{\partial n^2} + \kappa \frac{\partial\psi}{\partial n} + \frac{\partial^2\psi}{\partial s^2},$$

and the final correction formula is a simple Taylor series expansion

$$\begin{aligned} C_{\Delta}^{(\psi)} &= [\psi_{\bullet}] + \alpha \left[\frac{\partial\psi_{\bullet}}{\partial n} \right] + \frac{1}{2}\alpha^2 \left[\frac{\partial^2\psi_{\bullet}}{\partial n^2} \right] \\ &= [\psi_{\bullet}] + \alpha \left[\frac{\partial\psi_{\bullet}}{\partial n} \right] + \frac{1}{2}\alpha^2 \left([\omega_{\bullet}] - \kappa \left[\frac{\partial\psi_{\bullet}}{\partial n} \right] - \frac{\partial^2[\psi_{\bullet}]}{\partial s^2} \right) \end{aligned}$$

where α is the signed distance to the foot of perpendicular and κ is the local curvature at that point. More specifically, α is positive if the grid point is outside Γ (in Ω_+), and vice versa. The final correction to ω is then

$$\omega_{\diamond} \stackrel{\text{update}}{=} \omega_{\diamond} - \frac{C_{\Delta}^{(\psi)}}{h^2}.$$

After correcting the right hand side ω , we apply a fast Poisson solver called **Fishpack** to solve the Poisson equation $\Delta\psi = \omega$ and then obtain ψ immediately.

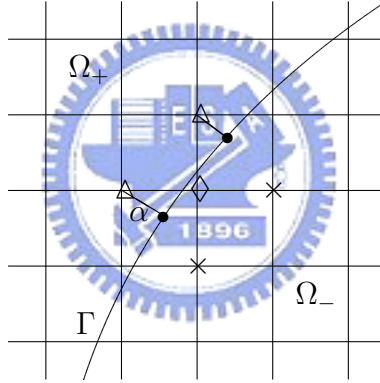


Figure 1: A 5-point Laplacian at \diamond . The two Δ are irregular points related to the irregular point \diamond .

2.3 Cubic Spline Representation

We interpret the interface Γ by picking some points on Γ , called *Lagrangian markers*, and using these markers to generate a *parameterized periodic cubic spline*. We denote the cubic spline by $\mathbf{X}(s) = (X(s), Y(s))$ where s , $0 < s < L$, is the *arc length parameter*. Hence, $|\partial\mathbf{X}/\partial s| = 1$ and then the local curvature can be expressed by $\kappa = X'Y'' - Y'X''$. The foot of perpendicular $\mathbf{X}(s_{i,j})$ of the grid point $\mathbf{x}_{i,j} = (x_i, y_j)$ can be found by *bisection method* just like root finding since the inner product of tangential and normal vector on a curve is equal to zero, That is,

$$(x_i - X(s_{i,j}))X'(s_{i,j}) + (y_j - Y(s_{i,j}))Y'(s_{i,j}) = 0.$$

Thus, we have the signed distance

$$\alpha = \pm \sqrt{(x_i - X(s_{i,j}))^2 + (y_j - Y(s_{i,j}))^2}, \quad (x_i, y_j) \in \Omega_{\pm}.$$

2.3.1 A Brief Introduction to Cubic Spline Representation

The cubic spline interpolation is significant since it yields C^2 approximation of curves and it is sufficiently smooth in the presence of small curvatures. In the construction of the periodic parametric cubic spline $\mathbf{X}(s)$, we give the markers

$$\mathbf{X}_k = (X_k, Y_k), \quad k = 1, 2, \dots, M$$

in order. We also define that $\mathbf{X}(s_k) = \mathbf{X}_k$, and impose that $\mathbf{X}_{M+1} = \mathbf{X}_1$ for convenience.

For $k = 1, 2, \dots, M$, We define the length of each segment by

$$h_{k+1} = \|\mathbf{X}_{k+1} - \mathbf{X}_k\|_2$$

and the accumulated length by

$$H_1 = 0, \quad H_{k+1} = H_k + h_{k+1}$$

at each marker, and then obtain the spline representation

$$\mathbf{X}(s) = \mathbf{X}_k + \mathbf{a}_1(s - H_k) + \mathbf{a}_2(s - H_k)^2 + \mathbf{a}_3(s - H_k)^3, \quad s \in [H_k, H_{k+1}]$$

where

$$\begin{aligned} \mathbf{a}_1 &= \frac{(\mathbf{X}_{k+1} - \mathbf{X}_k)}{h_{k+1}} - (2\mathbf{M}_k + \mathbf{M}_{k+1}) \frac{h_{k+1}}{6}, \\ \mathbf{a}_2 &= \frac{\mathbf{M}_k}{2}, \\ \mathbf{a}_3 &= \frac{(\mathbf{M}_{k+1} - \mathbf{M}_k)}{6h_{k+1}}. \end{aligned}$$

That is, the spline can be characterized by its *moments* $\mathbf{M}_i = \mathbf{X}''(s_i)$. In fact, we implement the parameterized cubic spline by using the *cumulative chord length* since the chord length is approximately equal to the arc length if the markers are close enough. For more detail, see [3] and [4].

By the continuity of $\mathbf{X}'(s)$ and periodicity of $\mathbf{X}(s)$, we have the moment equation

$$\mu_j \mathbf{M}_{j-1} + 2\mathbf{M}_j + \lambda_j \mathbf{M}_{j+1} = \mathbf{d}_j$$

$$\mathbf{M}_0 = \mathbf{M}_M, \quad \mathbf{M}_{M+1} = \mathbf{M}_1$$

for $j = 1, 2, \dots, M$, where

$$\begin{aligned} \lambda_j &= \frac{h_{j+1}}{h_j + h_{j+1}} \\ \mu_j &= \frac{h_j}{h_j + h_{j+1}} = 1 - \lambda_j \\ \mathbf{d}_j &= \frac{6}{h_j + h_{j+1}} \left(\frac{\mathbf{X}_{j+1} - \mathbf{X}_j}{h_{j+1}} - \frac{\mathbf{X}_j - \mathbf{X}_{j-1}}{h_j} \right) \end{aligned}$$

The task of calculating these moments is to solve a nonsingular linear system with a tridiagonal matrix containing two nonzero elements at the right-top and left-bottom corner.

Hence, we need to solve the nonsingular linear system

$$\begin{bmatrix} 2 & \lambda_1 & & & \mu_1 \\ \mu_2 & 2 & \lambda_2 & & \\ & \mu_3 & \ddots & \ddots & \\ & & \ddots & \ddots & \ddots \\ & & & \ddots & 2 & \lambda_{M-1} \\ \lambda_M & & & & \mu_M & 2 \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ \vdots \\ \vdots \\ \vdots \\ m_M \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ \vdots \\ \vdots \\ d_M \end{bmatrix}$$

for each component of $\mathbf{X}(s_j)$.

2.3.2 Solving the Linear System $Ax = b$ with Specific Matrix A

We introduce *Thomas algorithm*, a simplified form of Gaussian elimination that can be used to solve tridiagonal system of equations

$$c_j x_{j-1} + d_j x_j + e_j x_{j+1} = b_j, \quad j = 1, 2, \dots, n,$$

$$c_1 = 0, \quad e_n = 0.$$

in $\mathcal{O}(n)$. In this case

$$A = \begin{bmatrix} d_1 & e_1 & & & & \\ c_2 & d_2 & e_2 & & & \\ & c_3 & \ddots & \ddots & & \\ & & \ddots & \ddots & \ddots & \\ & & & \ddots & d_{n-1} & e_{n-1} \\ & & & & c_n & d_n \end{bmatrix}.$$

Thomas algorithm is given here:

```
!! Forward elimination phase
do k = 2,n
  m = c(k) / d(k-1)
  d(k) = d(k) - m*e(k-1)
  b(k) = b(k) - m*b(k-1)
end do
!! Backward substitution phase
x(n) = b(n)/d(n)
do k = n-1,1,-1
  x(k) = (b(k) - e(k)*x(k+1)) / d(k)
end do
```

For periodic case, i.e., system of equations gives that

$$c_j x_{j-1} + d_j x_j + e_j x_{j+1} = b_j, \quad j = 1, 2, \dots, n,$$

$$x_0 = x_1, \quad x_{n+1} = x_1$$

and

$$A = \begin{bmatrix} d_1 & e_1 & & & & c_1 \\ c_2 & d_2 & e_2 & & & \\ & c_3 & \ddots & \ddots & & \\ & & \ddots & \ddots & \ddots & \\ & & & \ddots & d_{n-1} & e_{n-1} \\ e_n & & & & c_n & d_n \end{bmatrix}.$$

We can solve the system by using Thomas algorithm twice.

1. Solve

$$c_j y_{j-1} + d_j y_j + e_j y_{j+1} = b_j, \quad j = 1, 2, \dots, n-1$$

with $y_0 = y_n = 0$

2. Solve

$$c_j z_{j-1} + d_j z_j + e_j z_{j+1} = 0, \quad j = 1, 2, \dots, n-1$$

with $z_0 = z_n = 1$

3. Obtaining

$$x_n = \frac{b_n - e_n y_1 - c_n y_{n-1}}{d_n - c_n z_{n-1} - e_n z_1},$$

and then we have

$$x_j = y_j + x_n z_j, \quad j = 1, 2, \dots, n-1.$$

A subroutine DGLTSL in Linpack can solve this periodic type of linear system in $\mathcal{O}(N)$, where N is the problem size.

2.3.3 Flag Construction

As we construct the spline, we also need to keep track of the irregular points. We allocate an integer array, called the *flag*, which denotes the Cartesian grid types. Since we assume that the interface is closed, the flag should record these types such as regular points inside/outside the interface and irregular points inside/outside/on the interface. By the way, the properties of the flag is somewhat like those of a *level set function*. We find the irregular point first by finding the neighbor point of the intersection of spline and grid line. Then, using some filling algorithm, the domain can be distinguished to be inside or outside. A basic pseudo code of filling algorithm is given here:

```
recursive subroutine FloodFill(ix, iy, oldColor, newColor)
  implicit none; integer, intent(in) :: ix, iy, oldColor, newColor
  if(IsOutside(ix,iy)) return
  if(iflag(ix,iy) == oldColor) then
    iflag(ix,iy) = newColor
    call FloodFill(ix+1, iy, oldColor, newColor)
    call FloodFill(ix-1, iy, oldColor, newColor)
    call FloodFill(ix, iy+1, oldColor, newColor)
    call FloodFill(ix, iy-1, oldColor, newColor)
  endif
  return
end subroutine FloodFill
```

Corresponding to the spline representation, the flag also plays an important role in the further computation.

2.4 Interpolation from Grid to Interface

Let $\psi_* = \psi(x_*, y_*)$ be the value on the interface which we want to interpolate. First we need to find the index (i, j) of grid point which is located on the left-bottom corner of the cell. Then, let $ph = x_* - x_i$, $qh = y_* - y_j$, with grid ratio $0 \leq p, q \leq 1$. Hence

$$\begin{aligned}\psi_+ &= p\psi_{i+1,j+1} + (1-p)\psi_{i,j+1} + \mathcal{O}(h^2) \\ \psi_\times &= p\psi_{i+1,j} + (1-p)\psi_{i,j} + \mathcal{O}(h^2) \\ \psi_* &= q\psi_+ + (1-q)\psi_\times + \mathcal{O}(h^2) \\ &= pq\psi_{i+1,j+1} + (1-p)q\psi_{i,j+1} + p(1-q)\psi_{i+1,j} + (1-p)(1-q)\psi_{i,j} + \mathcal{O}(h^2).\end{aligned}$$

The interpolation of an interface point from neighbor irregular points should be modified by correction terms. We define

$$\chi_{i,j} = \begin{cases} 1 & \mathbf{x}_{i,j} \in \Omega_+ \\ 0 & \mathbf{x}_{i,j} \in \bar{\Omega}_- \end{cases}$$

and we have

$$\psi_*^{\text{update}} = \psi_* + pq\chi_{i+1,j+1}C_{i+1,j+1}^{(\psi)} + (1-p)q\chi_{i,j+1}C_{i,j+1}^{(\psi)} + p(1-q)\chi_{i+1,j}C_{i+1,j}^{(\psi)} + (1-p)(1-q)\chi_{i,j}C_{i,j}^{(\psi)}.$$

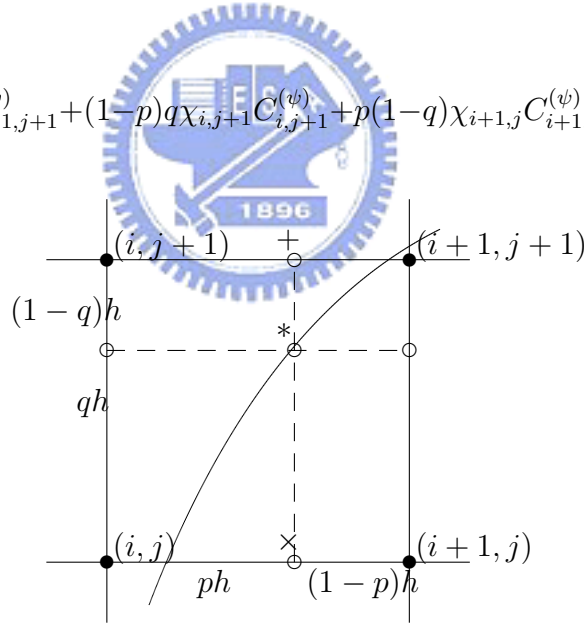


Figure 2: Interpolation on point * using the four corner points.

2.5 Numerical Example

We consider three modified Poisson equations with exact solutions on $\Omega = [-1, 1] \times [-1, 1]$ with jump discontinuities defined on a ellipse

$$\Gamma : \frac{x^2}{a^2} + \frac{y^2}{b^2} = 1.$$

We show two test cases: $a = 0.6, b = 0.4$, and $a = 0.8, b = 0.2$.

	Example. 1	Example. 2	Example. 3
ψ_-	1	$x^2 - y^2$	$\sin(x) \cos(y)$
ψ_+	$1 + \ln(2\sqrt{x^2 + y^2})$	0	0
ω_-	0	0	$-2 \sin(x) \cos(y)$
ω_+	0	0	0

Table 1: Three test cases for Eq. (2.1).

	N	$\ \cdot\ _\infty$	ratio	$\ \cdot\ _{L^2}$	ratio
Example. 1	16	3.0347E-02	0.0000	9.1019E-03	0.0000
	32	5.4561E-03	5.5621	6.4576E-04	14.0949
	64	1.7111E-03	3.1887	2.2393E-04	2.8838
	128	5.3475E-04	3.1997	1.6647E-04	1.3452
	256	8.7802E-05	6.0904	1.6368E-05	10.1703
Example. 2	16	7.2378E-02	0.0000	1.0009E-02	0.0000
	32	1.7418E-02	4.1553	1.3929E-03	7.1855
	64	2.7228E-03	6.3972	4.1537E-04	3.3535
	128	1.0008E-03	2.7206	1.3772E-04	3.0160
	256	1.0402E-04	9.6208	1.6698E-05	8.2479
Example. 3	16	4.2294E-02	0.0000	6.2970E-03	0.0000
	32	7.7546E-03	5.4540	7.7777E-04	8.0963
	64	1.2982E-03	5.9732	1.0272E-04	7.5721
	128	5.6115E-04	2.3135	1.1584E-04	0.8867
	256	1.4999E-04	3.7413	2.3262E-05	4.9799

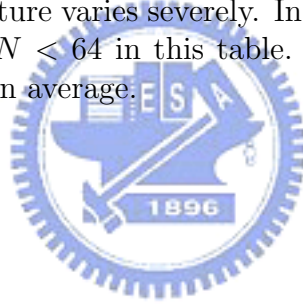
Table 2: Test results for $a = 0.6$, $b = 0.4$.

In the first case, the convergent rate fluctuates in a wide range. The accuracy may be affected by how the spline approximates the actual ellipse interface. General speaking, it has second order accuracy on average.

	N	$\ \cdot\ _\infty$	ratio	$\ \cdot\ _{L^2}$	ratio
Example. 1	16	2.1539E+00	0.0000	5.7049E-01	0.0000
	32	5.4730E-01	3.9355	9.5492E-02	5.9743
	64	1.2800E-02	42.7584	6.9241E-04	137.9125
	128	2.5374E-03	5.0446	1.9674E-04	3.5193
	256	9.1706E-04	2.7668	4.2801E-05	4.5967
Example. 2	16	3.9062E-01	0.0000	4.0670E-02	0.0000
	32	2.3950E-01	1.6310	4.0821E-02	0.9963
	64	2.1336E-02	11.2252	1.0262E-03	39.7800
	128	4.1392E-03	5.1545	3.9887E-04	2.5727
	256	1.4158E-03	2.9237	5.4272E-05	7.3495
Example. 2	16	5.6932E-01	0.0000	7.1246E-02	0.0000
	32	1.2497E-01	4.5556	1.3391E-02	5.3204
	64	8.8370E-03	14.1420	5.5884E-04	23.9622
	128	1.5545E-03	5.6848	9.7281E-05	5.7446
	256	5.4722E-04	2.8408	3.2040E-05	3.0363

Table 3: Test results for $a = 0.8$, $b = 0.2$.

In the second case, the curvature varies severely. In our test, the cartesian grid cannot fetch the interface behavior as $N < 64$ in this table. As N exceeds this value, we also observe second order accuracy on average.



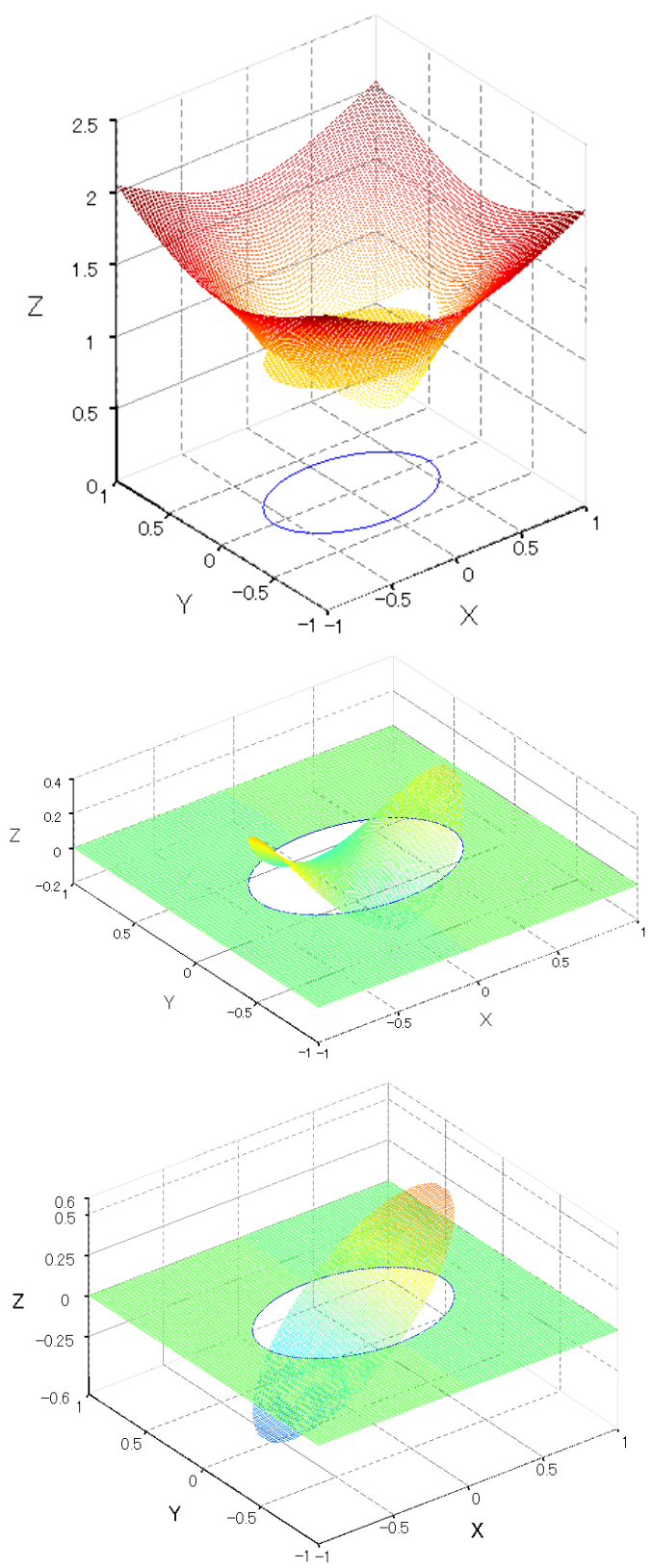


Figure 3: The results for the above examples, $a = 0.6, b = 0.4$.

3 Stokes Flow with Singular Force

3.1 Stokes Equation

The steady Stokes equation is of this form

$$\begin{aligned} -\nabla p + \mu \Delta \mathbf{u} &= \mathbf{f}, \\ \nabla \cdot \mathbf{u} &= 0. \end{aligned}$$

where $\mathbf{u} = (u(\mathbf{x}), v(\mathbf{x}))$ is the fluid velocity, $p = p(\mathbf{x})$ is the fluid pressure, and the μ is the constant fluid viscosity. The force density $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}))$ is applied by the immersed interface Γ into the fluid, which the boundary force is $\mathbf{F}(s) = (F_1(s), F_2(s))$. More clearly, the singular force \mathbf{f} has support only on Γ and has the form

$$\mathbf{f}(\mathbf{x}) = \int_{\Gamma} \mathbf{F}(s) \delta(\mathbf{x} - \mathbf{X}(s)) ds$$

where δ is the Dirac delta function defined in 2-D. The boundary configuration of Γ is given by $\mathbf{X}(s) = (X(s), Y(s))$, $0 \leq s \leq L$, where the are-length parameter s tracks every material point of the initial configuration.

We write a 2-D vector form of this equation, which yields

$$\begin{aligned} - \begin{bmatrix} p_x \\ p_y \end{bmatrix} + \mu \begin{bmatrix} \Delta u \\ \Delta v \end{bmatrix} &= \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}, \\ u_x + v_y &= 0. \end{aligned}$$

One can expect that the velocity and the pressure will not be smooth across the boundary. In fact the boundary force \mathbf{F} plays an important role to the jump condition of \mathbf{u} , p , and their derivatives. In the computational fluid dynamics, we usually impose the *no-slip condition* of fluid velocity, i.e., $[\mathbf{u}] = 0$ along the interface. Thus, we have the following theorem.

Theorem. Let \mathbf{u}, p be solution of Stokes' equation. Then the jump across the interface satisfies

$$[p] = \frac{\mathbf{F} \cdot \mathbf{n}}{|\partial \mathbf{X} / \partial s|}, \quad \mu \left[\frac{\partial \mathbf{u}}{\partial n} \right] = \frac{-\mathbf{F} \cdot \boldsymbol{\tau}}{|\partial \mathbf{X} / \partial s|} \boldsymbol{\tau}, \quad \left[\frac{\partial p}{\partial n} \right] = \frac{1}{|\frac{\partial \mathbf{X}}{\partial s}|} \frac{\partial}{\partial s} \left(\frac{\mathbf{F} \cdot \boldsymbol{\tau}}{|\partial \mathbf{X} / \partial s|} \right),$$

where \mathbf{n} , $\boldsymbol{\tau}$ are unit normal and tangent vector respectively on the interface.

We will give a proof in the next section.

3.2 Marker and Cell Method

We use *marker and cell* (MAC) method and take the computation domain Ω to be square (or rectangular) with staggered grid in Cartesian coordinates. For simplicity, we take the mesh size $h = \Delta x = \Delta y$ to be equal and let N_x , N_y be the number of mesh in the x - and y - directions respectively. Usually, we pick the same number of grid points in each direction, and denote that $N = N_x = N_y$.

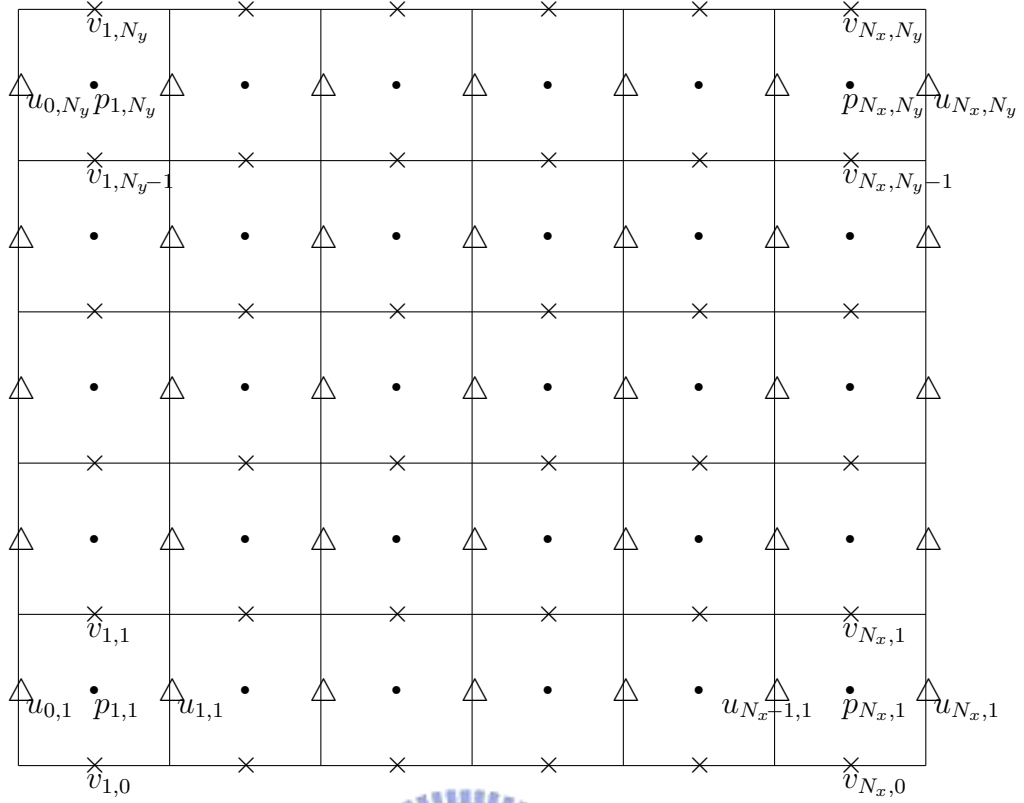


Figure 4: The computational domain Ω using staggered Grid with mesh size h .

3.3 Numerical Scheme

In the following numerical scheme, we assume s be an approximation of arc-length parameter such that $|\partial \mathbf{X} / \partial s|$ is nearly equal to 1. For simplicity, we denote $F_{\mathbf{n}} = \mathbf{F} \cdot \mathbf{n}$ and $F_{\boldsymbol{\tau}} = \mathbf{F} \cdot \boldsymbol{\tau}$. By translating the boundary force \mathbf{F} to the jump condition, we have

$$\begin{aligned}
 -\nabla p + \mu \Delta \mathbf{u} &= 0, \\
 \nabla \cdot \mathbf{u} &= 0, \\
 [\mathbf{u}] &= 0, \quad [p] = F_{\mathbf{n}}, \quad \mu \left[\frac{\partial \mathbf{u}}{\partial n} \right] = -F_{\boldsymbol{\tau}} \boldsymbol{\tau}, \quad \left[\frac{\partial p}{\partial n} \right] = \frac{\partial F_{\boldsymbol{\tau}}}{\partial s} \text{ on } \Gamma, \\
 \mathbf{u} \Big|_{\partial \Omega} &= \mathbf{u}_b.
 \end{aligned}$$

Consequently, by taking divergent to the above equations we have

$$\begin{aligned}
 \Delta p &= 0, \\
 [p] &= F_{\mathbf{n}}, \quad \left[\frac{\partial p}{\partial n} \right] = \frac{\partial F_{\boldsymbol{\tau}}}{\partial s} \text{ on } \Gamma.
 \end{aligned}$$

We need to impose a boundary condition for the above Laplace system. For the test example, we give the Dirichlet boundary condition with given solution p calculated on the boundary. In general, we consider the periodic boundary condition. Although any two solutions of this system may be up to a constant, it is fine since we need only ∇p .

Here we summarize the solution procedures.

1. Solve the Laplace equation

$$\Delta_h p + C^{(p)} = 0$$

and obtain p , where the correction term

$$\begin{aligned} C^{(p)} &= \frac{1}{h^2} \left([p] + \alpha \left[\frac{\partial p}{\partial n} \right] + \frac{1}{2} \alpha^2 \left([\Delta p] - \kappa \left[\frac{\partial p}{\partial n} \right] - \frac{\partial^2 [p]}{\partial s^2} \right) \right) \\ &= \frac{1}{h^2} \left(F_{\mathbf{n}} + \alpha \frac{\partial F_{\boldsymbol{\tau}}}{\partial s} + \frac{1}{2} \alpha^2 \left(-\kappa \frac{\partial F_{\boldsymbol{\tau}}}{\partial s} - \frac{\partial^2 F_{\mathbf{n}}}{\partial s^2} \right) \right). \end{aligned}$$

2. Compute p_x, p_y on the u - and v -grid by center difference $(p_{i+1,j} - p_{i,j})/h$, $(p_{i,j+1} - p_{i,j})/h$ respectively. It needs some correction when the p -grid points straddle across the interface. Let us explain the procedure more clearly by taking an example. Note that the location of $u_{i,j}$ is the middle point between those of $p_{i,j}$ and $p_{i,j}$. Let $p_{i+1,j} \in \Omega_-$ and $p_{i,j}, u_{i,j} \in \Omega_+$. The key point is to correct the point on the different side with the middle point. We need to add the correction term if the point is in Ω_- , and vice versa. In this case we need to add the correction term $C_{i+1,j}^{(p)}$ of $p_{i+1,j}$, which is defined as above. That is, the approximation of p_x can be expressed by $((p_{i+1,j} + C_{i+1,j}^{(p)}) - p_{i,j})/h$. We will discuss the detail in the next section.
3. Compute $[\nabla p]$ which appears in the correction term $\mathbf{C}^{(u)}$ later. Recall that $\boldsymbol{\tau} = (\tau_1, \tau_2)$, the unit tangent vector, is equal to $(-n_2, n_1)$ at each point on Γ , where $\mathbf{n} = (n_1, n_2)$ is the unit normal vector. Since

$$\begin{aligned} \frac{\partial F_{\boldsymbol{\tau}}}{\partial s} &= \left[\frac{\partial p}{\partial n} \right] = [p_x] n_1 + [p_y] n_2, \\ \frac{\partial F_{\mathbf{n}}}{\partial s} &= \frac{\partial [p]}{\partial s} = [p_x] \tau_1 + [p_y] \tau_2, \end{aligned}$$

we have

$$[\nabla p] = ([p_x], [p_y]) = \left(\tau_2 \frac{\partial F_{\boldsymbol{\tau}}}{\partial s} + \tau_1 \frac{\partial F_{\mathbf{n}}}{\partial s}, \quad -\tau_1 \frac{\partial F_{\boldsymbol{\tau}}}{\partial s} + \tau_2 \frac{\partial F_{\mathbf{n}}}{\partial s} \right).$$

4. Solve the Poisson equation

$$\mu \Delta_h \mathbf{u} + \mathbf{C}^{(u)} = \nabla_h p$$

where

$$\begin{aligned} \mathbf{C}^{(u)} &= \frac{1}{h^2} \left(\mu [\mathbf{u}] + \alpha \mu \left[\frac{\partial \mathbf{u}}{\partial n} \right] + \frac{1}{2} \alpha^2 \left(\mu [\Delta \mathbf{u}] - \kappa \mu \left[\frac{\partial \mathbf{u}}{\partial n} \right] - \mu \frac{\partial^2 [\mathbf{u}]}{\partial s^2} \right) \right) \\ &= \frac{1}{h^2} \left(-\alpha F_{\boldsymbol{\tau}} \boldsymbol{\tau} + \frac{1}{2} \alpha^2 ([\nabla p] + \kappa F_{\boldsymbol{\tau}} \boldsymbol{\tau}) \right). \end{aligned}$$

3.4 Handling the Derivatives with a Crossing Interface

Consider the derivatives approximated by center difference scheme with a crossing interface. In general, there are four cases about locations of grid points and the interface. We use a table to illustrate how to calculate the first order derivative at a certain point.

Case	Figure	Difference formula
(I)		$\frac{d\phi_M}{dx} \simeq \frac{(\phi_R + C_R) - \phi_L}{h}$
(II)		$\frac{d\phi_M}{dx} \simeq \frac{\phi_R - (\phi_L - C_L)}{h}$
(III)		$\frac{d\phi_M}{dx} \simeq \frac{(\phi_R - C_R) - \phi_L}{h}$
(IV)		$\frac{d\phi_M}{dx} \simeq \frac{\phi_R - (\phi_L + C_L)}{h}$

Table 4: The four cases about locations of grid points L, M, R and the interface. h denotes the length of the interval $[L, R]$, and M is the middle point in $[L, R]$ here. C_L, C_R are the correction terms at grid points L, R respectively.

3.5 Numerical Example

Example (1). Normal forces on a circle. Consider a circular boundary of radius $R = 1$ parametrized by $\mathbf{x}(\theta) = (\cos(\theta), \sin(\theta))$ in $\Omega = [-2, 2] \times [-2, 2]$ and define the force on the boundary by

$$\mathbf{f}(\theta) = 2 \sin(3\theta) \mathbf{x}(\theta)$$

so that the forces are normal to the boundary. The exact solution is given by

$$p(r, \theta) = \begin{cases} -r^3 \sin(3\theta), & r < 1 \\ r^{-3} \sin(3\theta), & r > 1 \end{cases}$$

$$u(r, \theta) = \begin{cases} \frac{3}{8}r^2 \sin(2\theta) + \frac{1}{16}r^4 \sin(4\theta) - \frac{1}{4}r^4 \sin(2\theta), & r < 1 \\ \frac{1}{8}r^{-2} \sin(2\theta) - \frac{3}{16}r^{-4} \sin(4\theta) + \frac{1}{4}r^{-2} \sin(4\theta), & r \geq 1 \end{cases}$$

$$v(r, \theta) = \begin{cases} \frac{3}{8}r^2 \cos(2\theta) - \frac{1}{16}r^4 \cos(4\theta) - \frac{1}{4}r^4 \cos(2\theta), & r < 1 \\ \frac{1}{8}r^{-2} \cos(2\theta) + \frac{3}{16}r^{-4} \cos(4\theta) - \frac{1}{4}r^{-2} \cos(4\theta), & r \geq 1 \end{cases}$$

We note that on the interface $R = 1$,

$$[p] = 2 \sin(3\theta), \quad \left[\frac{\partial p}{\partial n} \right] = 0, \quad [\Delta p] = 0, \quad \frac{\partial^2 [p]}{\partial s^2} = -18 \sin(3\theta)$$

and

$$[\mathbf{u}] = \mathbf{0}, \quad \left[\frac{\partial \mathbf{u}}{\partial n} \right] = \mathbf{0}, \quad [\Delta \mathbf{u}] = \begin{bmatrix} -6 \cos(3\theta) \sin(\theta) \\ 6 \cos(3\theta) \cos(\theta) \end{bmatrix}, \quad \frac{\partial^2 [\mathbf{u}]}{\partial s^2} = \mathbf{0}.$$

N	$\ u_{\text{err}}\ _{\infty}$	Ratio	$\ v_{\text{err}}\ _{\infty}$	Ratio	$\ p_{\text{err}}\ _{\infty}$	Ratio
16	6.31625996E-02	-	5.71511520E-02	-	1.80064548E-01	-
32	1.15154249E-02	5.4850	1.04312328E-02	5.4788	6.70713180E-02	2.6846
64	2.46761660E-03	4.6666	3.44416038E-03	3.0286	2.41626231E-02	2.7758
128	2.31427096E-04	10.6626	6.66326129E-04	5.1688	3.79474345E-03	6.3673

Table 5: A grid refinement analysis of u , v , p for Example(1).

Example (2). Tangential forces on a circle. We consider a similar case. The domain Ω and the circular boundary are defined as Example(1). Define the force on the boundary by

$$\mathbf{f}(\theta) = 2 \sin(3\theta) \mathbf{x}'(\theta)$$

so that the forces are tangential to the boundary. The exact solution is given by

$$p(r, \theta) = \begin{cases} -r^3 \cos(3\theta), & r < 1 \\ -r^{-3} \cos(3\theta), & r > 1 \end{cases}$$

$$u(r, \theta) = \begin{cases} \frac{1}{8}r^2 \cos(2\theta) + \frac{1}{16}r^4 \cos(4\theta) - \frac{1}{4}r^4 \cos(2\theta), & r < 1 \\ -\frac{1}{8}r^{-2} \cos(2\theta) + \frac{5}{16}r^{-4} \cos(4\theta) - \frac{1}{4}r^{-2} \cos(4\theta), & r \geq 1 \end{cases}$$

$$v(r, \theta) = \begin{cases} -\frac{1}{8}r^2 \sin(2\theta) + \frac{1}{16}r^4 \sin(4\theta) + \frac{1}{4}r^4 \sin(2\theta), & r < 1 \\ \frac{1}{8}r^{-2} \sin(2\theta) + \frac{5}{16}r^{-4} \sin(4\theta) - \frac{1}{4}r^{-2} \sin(4\theta), & r \geq 1 \end{cases}$$

We note that on the interface $R = 1$,

$$[p] = 0, \quad \left[\frac{\partial p}{\partial n} \right] = 6 \cos(3\theta), \quad [\Delta p] = 0, \quad \frac{\partial^2 [p]}{\partial s^2} = 0$$

and

$$[\mathbf{u}] = \mathbf{0}, \quad \left[\frac{\partial \mathbf{u}}{\partial n} \right] = \begin{bmatrix} 2 \sin(3\theta) \sin(\theta) \\ -2 \sin(3\theta) \cos(\theta) \end{bmatrix}, \quad [\Delta \mathbf{u}] = \begin{bmatrix} 3 \cos(2\theta) + 3 \cos(4\theta) \\ -3 \sin(2\theta) + 3 \sin(4\theta) \end{bmatrix}, \quad \frac{\partial^2 [\mathbf{u}]}{\partial s^2} = \mathbf{0}.$$

N	$\ u_{\text{err}}\ _{\infty}$	Ratio	$\ v_{\text{err}}\ _{\infty}$	Ratio	$\ p_{\text{err}}\ _{\infty}$	Ratio
16	5.56812172E-02	-	1.84446032E-02	-	1.92715009E-01	-
32	9.06017465E-03	6.1457	5.85585435E-03	3.1497	2.08500578E-02	9.2429
64	3.34797871E-03	2.7061	1.43026464E-03	4.0942	5.88284456E-03	3.5442
128	6.04375427E-04	5.5395	3.19944420E-04	4.4703	1.32162086E-03	4.4512

Table 6: A grid refinement analysis of u , v , p for Example(2).

Example (3). Combine the normal and tangential forces on a circle. Since the Stokes equation is linear, we combine the above two example by taking summation.

N	$\ u_{\text{err}}\ _{\infty}$	Ratio	$\ v_{\text{err}}\ _{\infty}$	Ratio	$\ p_{\text{err}}\ _{\infty}$	Ratio
16	1.17188492E-01	-	6.20299174E-02	-	2.02998893E-01	-
32	1.22218170E-02	9.5884	1.25574945E-02	4.9396	8.55532629E-02	2.3727
64	3.73073530E-03	3.2759	3.67928831E-03	3.4130	2.61792711E-02	3.2679
128	5.23096557E-04	7.1320	7.33869339E-04	5.0135	4.55700360E-03	5.7448

Table 7: A grid refinement analysis of u , v , p for Example(3).

Example (4). A moving interface problem. This example is given in [8]. We consider a moving interface problem which has uniform density and viscosity. The initial interface is given by

$$r(\theta) = r_0(1 + \epsilon \sin(k\theta)), \quad 0 \leq \theta \leq 2\pi.$$

The initial velocity and pressure are all set to be zero. The only driven force of the fluid motion is the surface tension which is proportional to the curvature κ with a constant E , that is,

$$F_{\mathbf{n}} = -E\kappa, \quad F_{\boldsymbol{\tau}} = 0.$$

The velocity is smooth but the pressure has a non-constant jump

$$[p] = F_{\mathbf{n}}, \quad \left[\frac{\partial p}{\partial n} \right] = 0.$$

We simply take an interface moving formula which says

$$\mathbf{X}^{n+1} = \mathbf{X}^n + \mathbf{U}^{n+1} \Delta t.$$

In our test, we take $r_0 = 0.5$, $k = 5$, $\epsilon = 0.2$, $E = 0.05$ and $\mu = 1$.

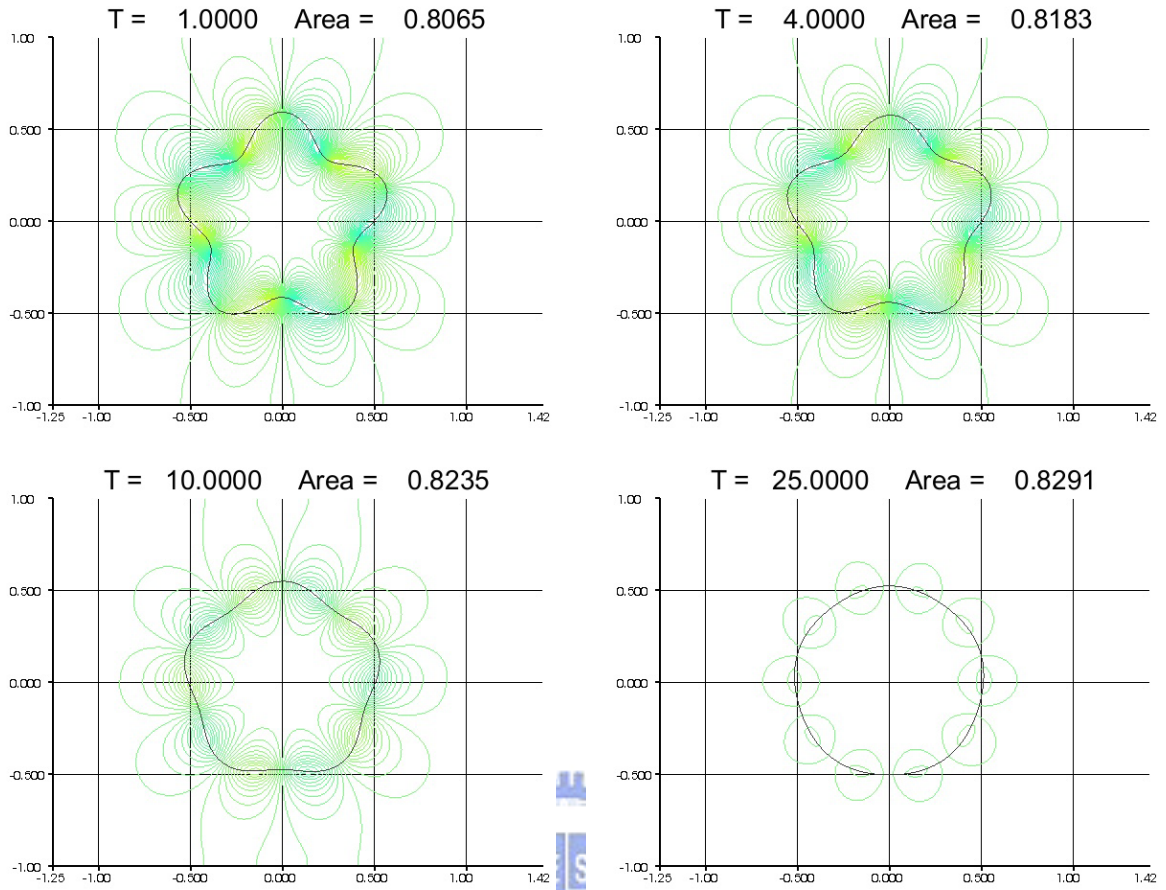


Figure 5: The numerical evolution of the moving interface using the IIM coupled with the interpolation formula introduced in previous section. Here we pick the number of grid: $N_x = N_y = N = 64$, number of interface marker: $M = 42$, and $\Delta t = 0.002$.

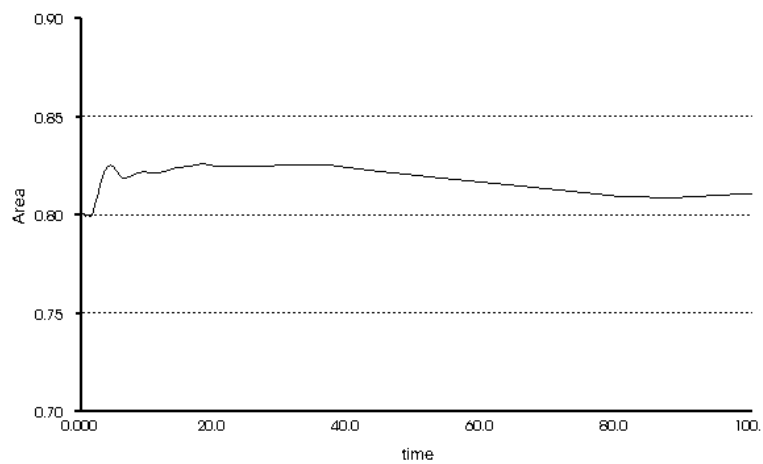


Figure 6: The measure of the bubble area. The geometry of interface eventually converges to a circle with conserving measure of area.

4 Navier-Stokes Equation

4.1 Introduction to Navier-Stokes Equation

The Navier Stokes Equation is of this form

$$\rho \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right) + \nabla p = \mu \Delta \mathbf{u} + \mathbf{f} + \mathbf{g}, \quad (4.2)$$

$$\nabla \cdot \mathbf{u} = 0, \quad (4.3)$$

where $\mathbf{u}(\mathbf{x}, t)$, $p(\mathbf{x}, t)$, μ , $\mathbf{f}(\mathbf{x}, t)$ are similar as the above session, and $\mathbf{g}(\mathbf{x}, t)$ is the external force defined on the computational domain. Note that the divergent-free condition $\nabla \cdot \mathbf{u} = 0$ denotes the fluid is incompressible.

Here we have some important theorems for Navier-Stokes equation, which can also be applied to time-independent Stokes equation.

Lemma. If $[\mathbf{u}] = 0$, then

$$[\mathbf{u}_t] + [\nabla \mathbf{u}] \cdot \mathbf{u} = 0.$$

Proof.

$$0 = \frac{d}{dt} [\mathbf{u}] = \left[\mathbf{u}_t + \nabla \mathbf{u} \cdot \frac{\partial \mathbf{X}}{\partial t} \right] = [\mathbf{u}_t + \nabla \mathbf{u} \cdot \mathbf{u}] = [\mathbf{u}_t] + [\nabla \mathbf{u}] \cdot \mathbf{u}.$$

□

Lemma. If $[\mathbf{u}] = 0$ and $\nabla \cdot \mathbf{u} = 0$, then

$$\left[\frac{\partial \mathbf{u}}{\partial n} \right] \cdot \mathbf{n} = 0.$$

Proof. Denoting the unit normal and tangential vector be $\mathbf{n} = (n_1, n_2)$, $\boldsymbol{\tau} = (-n_2, n_1)$, and imposing $u_x + v_y = 0$, we have

$$\begin{aligned} \left[\frac{\partial \mathbf{u}}{\partial n} \right] \cdot \mathbf{n} &= [u_x n_1 + u_y n_2] n_1 + [v_x n_1 + v_y n_2] n_2 \\ &= [u_x n_1 + v_x n_2] n_1 + [u_y n_1 + v_y n_2] n_2 \\ &= [v_y (-n_1) + v_x n_2] n_1 + [u_y n_1 + u_x (-n_2)] n_2 \\ &= -[v_y n_1 - v_x n_2] n_1 + [u_y n_1 - u_x n_2] n_2 \\ &= - \left[\frac{\partial \mathbf{u}}{\partial \boldsymbol{\tau}} \right] \cdot \boldsymbol{\tau} = 0 \end{aligned}$$

since

$$\left[\frac{\partial \mathbf{u}}{\partial \boldsymbol{\tau}} \right] = \frac{\partial [\mathbf{u}]}{\partial \boldsymbol{\tau}} = 0.$$

□

Theorem. Let \mathbf{u}, p be solution of Navier-Stokes equation without the external force \mathbf{g} . Then the jump across the interface satisfies

$$[p] = \frac{\mathbf{F} \cdot \mathbf{n}}{|\partial \mathbf{X} / \partial s|}, \quad \mu \left[\frac{\partial \mathbf{u}}{\partial n} \right] = \frac{-\mathbf{F} \cdot \boldsymbol{\tau}}{|\partial \mathbf{X} / \partial s|} \boldsymbol{\tau}, \quad \left[\frac{\partial p}{\partial n} \right] = \frac{1}{|\frac{\partial \mathbf{X}}{\partial s}|} \frac{\partial}{\partial s} \left(\frac{\mathbf{F} \cdot \boldsymbol{\tau}}{|\partial \mathbf{X} / \partial s|} \right)$$

where $\mathbf{n}, \boldsymbol{\tau}$ are unit normal and tangent vector respectively on the interface Γ .

Proof. First we choose a banded domain $\Omega_\epsilon(t)$ enclosing the interface $\Gamma(t)$ with the outer and inner boundary $\Gamma_\epsilon^+(t)$ and $\Gamma_\epsilon^-(t)$. The ϵ is a small parameter of the distance from $\Gamma(t)$ to $\Gamma_\epsilon^+(t)$ and to $\Gamma_\epsilon^-(t)$, where $\Gamma(t) = \{\mathbf{X}(s, t), 0 \leq s \leq L\}$ be the parametric curve with parameter s and dependence on time t . Let $\phi(\mathbf{x})$ be any smooth function with compact support defined on $\Omega_\epsilon(t)$. We multiply $\phi(\mathbf{x})$ on both sides of Navier-Stokes equation and integrate over $\Omega_\epsilon(t)$, which yields

$$\begin{aligned} & \int_{\Omega_\epsilon(t)} \rho \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) \phi \, d\mathbf{x} + \int_{\Omega_\epsilon(t)} \nabla p \phi \, d\mathbf{x} \\ &= \int_{\Omega_\epsilon(t)} \mu \Delta \mathbf{u} \phi \, d\mathbf{x} + \int_{\Omega_\epsilon(t)} \int_0^L \mathbf{F}(s, t) \delta(\mathbf{x} - \mathbf{X}(s, t)) \, ds \, \phi \, d\mathbf{x}. \end{aligned}$$

Using the relation of the transport theorem in the first term of the left hand side, and the fact of $\Omega_\epsilon(t)$ enclosing $\Gamma(t)$ and the definition of Dirac delta function in the last term of right hand side, we have

$$\frac{d}{dt} \int_{\Omega_\epsilon(t)} \rho \mathbf{u} \phi \, d\mathbf{x} + \int_{\Omega_\epsilon(t)} \nabla p \phi \, d\mathbf{x} = \int_{\Omega_\epsilon(t)} \mu \Delta \mathbf{u} \phi \, d\mathbf{x} + \int_0^L \mathbf{F}(s, t) \phi(\mathbf{X}(s, t)) \, ds.$$

Let us consider the first component of the above equation, that is,

$$\frac{d}{dt} \int_{\Omega_\epsilon(t)} \rho u \phi \, d\mathbf{x} + \int_{\Omega_\epsilon(t)} p_x \phi \, d\mathbf{x} = \int_{\Omega_\epsilon(t)} \mu \Delta u \phi \, d\mathbf{x} + \int_0^L F_1(s, t) \phi(\mathbf{X}(s, t)) \, ds.$$

Assume u is smooth enough and bounded, the first term of left hand side tends to zero as $\epsilon \rightarrow 0$. Moreover

$$\begin{aligned} \int_{\Omega_\epsilon(t)} p_x \phi \, d\mathbf{x} &= \int_{\Gamma_\epsilon^+} p n_1 \phi \, d\sigma - \int_{\Gamma_\epsilon^-} p n_1 \phi \, d\sigma - \int_{\Omega_\epsilon(t)} p \phi_x \, d\mathbf{x} \\ &\rightarrow \int_\Gamma [p] n_1 \phi \, d\sigma = \int_0^L [p] n_1 \phi |\partial \mathbf{X} / \partial s| \, ds \end{aligned}$$

and

$$\begin{aligned} \int_{\Omega_\epsilon(t)} \mu \Delta u \phi \, d\mathbf{x} &= \int_{\Gamma_\epsilon^+} \mu \frac{\partial u}{\partial n} \phi \, d\sigma - \int_{\Gamma_\epsilon^-} \mu \frac{\partial u}{\partial n} \phi \, d\sigma - \int_{\Omega_\epsilon(t)} \mu \nabla u \cdot \nabla \phi \, d\mathbf{x} \\ &\rightarrow \int_\Gamma \mu \left[\frac{\partial u}{\partial n} \right] \phi \, d\sigma = \int_0^L \mu \left[\frac{\partial u}{\partial n} \right] \phi |\partial \mathbf{X} / \partial s| \, ds \end{aligned}$$

as $\epsilon \rightarrow 0$. Therefore, since ϕ is arbitrary, we have

$$[p] |\partial \mathbf{X} / \partial s| n_1 = \mu \left[\frac{\partial u}{\partial n} \right] |\partial \mathbf{X} / \partial s| + F_1.$$

Similarly, for the second component, we have

$$[p] |\partial \mathbf{X} / \partial s| n_2 = \mu \left[\frac{\partial u}{\partial n} \right] |\partial \mathbf{X} / \partial s| + F_2.$$

Multiplying n_1, n_2 on both sides of the above equations respectively, summing them, and by previous lemma, we obtain the first equality

$$[p] = \frac{\mathbf{F} \cdot \mathbf{n}}{|\partial \mathbf{X} / \partial s|}.$$

Now substituting this pressure jump into the above relations, we get

$$\begin{aligned}\mu \left[\frac{\partial u}{\partial n} \right] | \partial \mathbf{X} / \partial s | &= (\mathbf{F} \cdot \mathbf{n}) n_1 - F_1 = (-F_1 n_2 + F_2 n_1) n_2 = (-\mathbf{F} \cdot \boldsymbol{\tau})(-n_2), \\ \mu \left[\frac{\partial v}{\partial n} \right] | \partial \mathbf{X} / \partial s | &= (\mathbf{F} \cdot \mathbf{n}) n_2 - F_2 = (F_1 n_2 - F_2 n_1) n_1 = (-\mathbf{F} \cdot \boldsymbol{\tau}) n_1,\end{aligned}$$

that is,

$$\mu \left[\frac{\partial \mathbf{u}}{\partial n} \right] = \frac{-\mathbf{F} \cdot \boldsymbol{\tau}}{|\partial \mathbf{X} / \partial s|} \boldsymbol{\tau}.$$

In order to derive the normal derivative jump of the pressure, let us apply the divergence operator to both side of Navier-Stokes equation and use the incompressibility constraint, we have

$$\nabla \cdot \left(\rho \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right) \right) + \Delta p = \nabla \cdot \mathbf{f}.$$

As before, we multiply ϕ on both sides and integrate over Ω_ϵ , which yields

$$\int_{\Omega_\epsilon(t)} \nabla \cdot \left(\rho \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right) \right) \phi \, d\mathbf{x} + \int_{\Omega_\epsilon(t)} \Delta p \phi \, d\mathbf{x} = \int_{\Omega_\epsilon(t)} \nabla \cdot \mathbf{f} \phi \, d\mathbf{x}.$$

Observe that

$$\begin{aligned}& \int_{\Omega_\epsilon(t)} \nabla \cdot \left(\rho \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right) \right) \phi \, d\mathbf{x} \\ &= \int_{\Omega_\epsilon(t)} \nabla \cdot \left(\rho \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right) \phi \right) \, d\mathbf{x} - \int_{\Omega_\epsilon(t)} \rho \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right) \cdot \nabla \phi \, d\mathbf{x} \\ &= \int_{\Gamma_\epsilon^+ \cup \Gamma_\epsilon^-} \rho \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right) \phi \cdot \mathbf{n} \, d\sigma - \frac{d}{dt} \int_{\Omega_\epsilon(t)} \rho \mathbf{u} \cdot \nabla \phi \, d\mathbf{x} \\ &\rightarrow 0 \quad (\text{as } \epsilon \rightarrow 0, \text{ by previous lemma, and } \mathbf{u} \text{ is bounded.})\end{aligned}$$

and, using twice the Green's identity,

$$\begin{aligned}\int_{\Omega_\epsilon(t)} \Delta p \phi \, d\mathbf{x} &= \int_{\Omega_\epsilon(t)} \Delta \phi p \, d\mathbf{x} + \int_{\Gamma_\epsilon^+ \cup \Gamma_\epsilon^-} \phi \frac{\partial p}{\partial n} \, d\sigma - \int_{\Gamma_\epsilon^+ \cup \Gamma_\epsilon^-} p \frac{\partial \phi}{\partial n} \, d\sigma \\ &\rightarrow \int_\Gamma \left[\frac{\partial p}{\partial n} \right] \phi \, d\sigma - \int_\Gamma [p] \frac{\partial \phi}{\partial n} \, d\sigma \quad (\text{as } \epsilon \rightarrow 0, \text{ and } p \text{ is bounded.}) \\ &= \int_0^L \left[\frac{\partial p}{\partial n} \right] | \partial \mathbf{X} / \partial s | \phi \, ds - \int_0^L [p] | \partial \mathbf{X} / \partial s | \frac{\partial \phi}{\partial n} \, ds.\end{aligned}$$

Using the definition of the dipole distribution (the generalized derivation of the Dirac

delta function), we have

$$\begin{aligned}
\int_{\Omega_\epsilon(t)} \nabla \cdot \mathbf{f} \phi \, d\mathbf{x} &= \int_{\Omega_\epsilon(t)} \int_0^L \left(\nabla \cdot (\mathbf{F} \delta(\mathbf{x} - \mathbf{X}(s, t))) \right) \phi \, ds d\mathbf{x} \\
&= \int_0^L \mathbf{F}(s, t) \left(\int_{\Omega_\epsilon(t)} (\nabla \cdot \delta(\mathbf{x} - \mathbf{X}(s, t))) \phi \, d\mathbf{x} \right) ds \\
&= - \int_0^L \mathbf{F} \cdot \nabla \phi \, ds \\
&= - \int_0^L \mathbf{F} \cdot \mathbf{n} \frac{\partial \phi}{\partial n} \, ds - \int_0^L \mathbf{F} \cdot \boldsymbol{\tau} \frac{\partial \phi}{\partial \tau} \, ds \\
&= - \int_0^L \mathbf{F} \cdot \mathbf{n} \frac{\partial \phi}{\partial n} \, ds - \int_0^L \frac{\mathbf{F} \cdot \boldsymbol{\tau}}{|\partial \mathbf{X} / \partial s|} \frac{\partial \phi}{\partial s} \, ds \\
&= - \int_0^L [p] |\partial \mathbf{X} / \partial s| \frac{\partial \phi}{\partial n} \, ds + \int_0^L \frac{\partial}{\partial s} \left(\frac{\mathbf{F} \cdot \boldsymbol{\tau}}{|\partial \mathbf{X} / \partial s|} \right) \phi \, ds.
\end{aligned}$$

Since ϕ is arbitrary, we obtain the normal derivative jump for the pressure

$$\left[\frac{\partial p}{\partial n} \right] = \frac{1}{|\frac{\partial \mathbf{X}}{\partial s}|} \frac{\partial}{\partial s} \left(\frac{\mathbf{F} \cdot \boldsymbol{\tau}}{|\partial \mathbf{X} / \partial s|} \right).$$

Thus, the proof is complete. □

Remark. The results can be generalized to the case with additional bounded piecewise continuous force \mathbf{g} across the interface. The proof is straightforward without much effort needed, and the result of $[\mathbf{u}]$, $[\frac{\partial \mathbf{u}}{\partial n}]$ are still valid. However, the normal derivative jump condition for the pressure should be

$$\left[\frac{\partial p}{\partial n} \right] = \frac{1}{|\frac{\partial \mathbf{X}}{\partial s}|} \frac{\partial}{\partial s} \left(\frac{\mathbf{F} \cdot \boldsymbol{\tau}}{|\partial \mathbf{X} / \partial s|} \right) + [\mathbf{g}] \cdot \mathbf{n}.$$

We consider the fluid density $\rho \equiv 1$, and translate the interface force to the jump conditions. We have

$$\begin{aligned}
\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} + \nabla p &= \mu \Delta \mathbf{u} + \mathbf{g}, \\
\nabla \cdot \mathbf{u} &= 0, \\
[\mathbf{u}] &= \mathbf{0}, \quad \mu \left[\frac{\partial \mathbf{u}}{\partial n} \right] = \frac{-\mathbf{F} \cdot \boldsymbol{\tau}}{|\partial \mathbf{X} / \partial s|} \boldsymbol{\tau}, \\
[p] &= \frac{\mathbf{F} \cdot \mathbf{n}}{|\partial \mathbf{X} / \partial s|}, \quad \left[\frac{\partial p}{\partial n} \right] = \frac{1}{|\frac{\partial \mathbf{X}}{\partial s}|} \frac{\partial}{\partial s} \left(\frac{\mathbf{F} \cdot \boldsymbol{\tau}}{|\partial \mathbf{X} / \partial s|} \right).
\end{aligned}$$

4.2 Numerical Scheme

There are 3 steps to approximate the solution numerically. Here we consider the scheme without interfaces first.

1 In the prediction step, we solve \mathbf{u}^* explicitly where

$$\frac{\mathbf{u}^* - \mathbf{u}^n}{\Delta t} + (\mathbf{u} \cdot \nabla) \mathbf{u}^{n+\frac{1}{2}} + \nabla p^{n-\frac{1}{2}} = \frac{\mu}{2} \Delta \mathbf{u}^n + \mathbf{g}^{n+\frac{1}{2}}, \quad (4.4)$$

and $\tilde{\mathbf{u}}^{n+1}$ by a fast Poisson solver with jump discontinuity,

$$\begin{aligned} \frac{\tilde{\mathbf{u}}^{n+1} - \mathbf{u}^*}{\Delta t} &= \frac{\mu}{2} \Delta \tilde{\mathbf{u}}^{n+1}, \\ \tilde{\mathbf{u}}^{n+1} \Big|_{\partial\Omega} &= \mathbf{u}^{n+1} \Big|_{\partial\Omega}, \\ [\tilde{\mathbf{u}}^{n+1}] &= 0, \quad \mu \left[\frac{\partial \tilde{\mathbf{u}}^{n+1}}{\partial n} \right] = \frac{-\mathbf{F}^* \cdot \boldsymbol{\tau}}{|\partial \mathbf{X} / \partial s|} \boldsymbol{\tau}. \end{aligned} \quad (4.5)$$

Note that the advection term $(\mathbf{u} \cdot \nabla) \mathbf{u}^{n+\frac{1}{2}}$ is approximated by the extrapolation

$$\frac{3}{2}(\mathbf{u} \cdot \nabla) \mathbf{u}^n - \frac{1}{2}(\mathbf{u} \cdot \nabla) \mathbf{u}^{n-1}.$$

We explain this step in detail. In the first component of the vector field we have these terms of the space discretization

$$\begin{aligned} \left(u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} \right)^{n+\frac{1}{2}} &= \frac{3}{2} \left(u_{i,j}^n \frac{u_{i+1,j}^n - u_{i-1,j}^n}{2\Delta x} + v_{stag}^n \frac{u_{i,j+1}^n - u_{i,j-1}^n}{2\Delta y} \right) \\ &\quad - \frac{1}{2} \left(u_{i,j}^{n-1} \frac{u_{i+1,j}^{n-1} - u_{i-1,j}^{n-1}}{2\Delta x} + v_{stag}^{n-1} \frac{u_{i,j+1}^{n-1} - u_{i,j-1}^{n-1}}{2\Delta y} \right), \\ \frac{\partial p^{n-\frac{1}{2}}}{\partial x} &= \frac{p_{i+1,j}^{n-\frac{1}{2}} - p_{i,j}^{n-\frac{1}{2}}}{\Delta x}, \\ \Delta u^n &= \frac{u_{i+1,j}^n - 2u_{i,j}^n + u_{i-1,j}^n}{\Delta x^2} + \frac{u_{i,j+1}^n - 2u_{i,j}^n + u_{i,j-1}^n}{\Delta y^2}, \end{aligned}$$

and the interpolation

$$v_{stag} = \frac{1}{4} (v_{i,j} + v_{i,j+1} + v_{i+1,j} + v_{i+1,j-1})$$

from v -grid to u -grid. The second component is similar.

- 2 We perform the projection step to correction the velocity to be divergent free and update the pressure. According to Helmholtz-Hodge decomposition, we have

$$\begin{aligned} \tilde{\mathbf{u}}^{n+1} &= \mathbf{u}^{n+1} + \Delta t \nabla \phi^{n+1}, \\ \nabla \cdot \mathbf{u}^{n+1} &= 0. \end{aligned} \quad (4.6)$$

That is, we want to project the vector field $\tilde{\mathbf{u}}^{n+1}$ to a divergent free one, \mathbf{u}^{n+1} . Taking divergent on both sides, we obtain

$$\begin{aligned} \frac{\nabla \cdot \tilde{\mathbf{u}}^{n+1}}{\Delta t} &= \Delta \phi^{n+1}, \\ \frac{\partial \phi^{n+1}}{\partial n} \Big|_{\partial\Omega} &= 0. \end{aligned}$$

Note that we should calculate the divergent term $\nabla \cdot \mathbf{u}^*$ on the p -grid, solve the Laplace equation of ϕ , and then obtain \mathbf{u}^{n+1} .

- 3 Finally, the pressure should be corrected. See [6] which says that

$$\nabla p^{n+\frac{1}{2}} = \nabla p^{n-\frac{1}{2}} + \nabla \phi^{n+1} - \frac{\mu \Delta t}{2} \nabla (\Delta \phi^{n+1}).$$

Similarly,

$$p^{n+\frac{1}{2}} = p^{n-\frac{1}{2}} + \phi^{n+1} - \frac{\mu}{2} \nabla \cdot \tilde{\mathbf{u}}^{n+1}.$$

It can be obtained by substituting (4.6) into the summation of (4.4) and (4.5).

4.3 Numerical Example

4.3.1 Accuracy Check without Interfaces

Here we introduce some example with exact solutions. In the following examples, we put the time marching step $\Delta t = h/2$ where h is again the mesh size of the staggered grid.

Example (1). Consider the Navier-Stokes equation on a domain $\Omega = [0, 1] \times [0, 1]$, and give the exact solution

$$\begin{aligned} u(x, y, t) &= -\cos(\pi x) \sin(\pi y) \exp(-2\pi^2 t / \text{Re}), \\ v(x, y, t) &= \sin(\pi x) \cos(\pi y) \exp(-2\pi^2 t / \text{Re}), \\ p(x, y, t) &= -0.25 (\cos(2\pi x) + \cos(2\pi y)) \exp(-4\pi^2 t / \text{Re}). \end{aligned}$$

where $\text{Re} = 1/\mu$. We have

$$\begin{aligned} \frac{\partial u}{\partial x} &= \sin(\pi x) \pi \sin(\pi y) \exp(-2\pi^2 t / \text{Re}), & \frac{\partial u}{\partial y} &= -\cos(\pi x) \cos(\pi y) \pi \exp(-2\pi^2 t / \text{Re}), \\ \frac{\partial^2 u}{\partial x^2} &= \cos(\pi x) \pi^2 \sin(\pi y) \exp(-2\pi^2 t / \text{Re}), & \frac{\partial^2 u}{\partial y^2} &= \cos(\pi x) \pi^2 \sin(\pi y) \exp(-2\pi^2 t / \text{Re}), \\ \frac{\partial v}{\partial x} &= \cos(\pi x) \cos(\pi y) \pi \exp(-2\pi^2 t / \text{Re}), & \frac{\partial v}{\partial y} &= -\sin(\pi x) \pi \sin(\pi y) \exp(-2\pi^2 t / \text{Re}), \\ \frac{\partial^2 v}{\partial x^2} &= -\sin(\pi x) \pi^2 \cos(\pi y) \exp(-2\pi^2 t / \text{Re}), & \frac{\partial^2 v}{\partial y^2} &= -\sin(\pi x) \pi^2 \cos(\pi y) \exp(-2\pi^2 t / \text{Re}), \\ \frac{\partial p}{\partial x} &= \frac{1}{2} \sin(2\pi x) \pi \exp(-4\pi^2 t / \text{Re}), & \frac{\partial p}{\partial y} &= \frac{1}{2} \sin(2\pi y) \pi \exp(-4\pi^2 t / \text{Re}). \end{aligned}$$

N	$\ u_{\text{error}}\ _2$	Ratio	$\ v_{\text{error}}\ _2$	Ratio	$\ p_{\text{error}}\ _2$	Ratio
16	2.5618E-05	-	1.7061E-05	-	6.1794E-03	-
32	2.7016E-06	9.4825	1.5956E-06	10.6925	1.2137E-03	5.0913
64	2.8951E-07	9.3316	1.7090E-07	9.3364	2.6025E-04	4.6635
128	3.5376E-08	8.1837	2.0645E-08	8.2780	6.0763E-05	4.2830

Table 8: A grid refinement analysis in L^2 -Norm for Example(1).

N	$\ u_{\text{error}}\ _\infty$	Ratio	$\ v_{\text{error}}\ _\infty$	Ratio	$\ p_{\text{error}}\ _\infty$	Ratio
16	8.2522E-05	-	6.6571E-05	-	1.0643E-02	-
32	1.5195E-05	5.4308	9.4561E-06	7.0400	2.1667E-03	4.9120
64	2.4185E-06	6.2828	1.4674E-06	6.4441	4.8205E-04	4.4947
128	4.2419E-07	5.7014	2.4059E-07	6.0991	1.1171E-04	4.3151

Table 9: A grid refinement analysis in L^∞ -Norm for Example(1).

Example (2). Consider the Navier-Stokes equation with $\mu = 1$ on a domain $\Omega = [0, 1] \times$

$[0, 1]$, and give the exact solution

$$\begin{aligned}
u(x, y, t) &= \frac{1}{4}ye^{-t}, \\
v(x, y, t) &= -\frac{1}{4}x(1-x^2)e^{-t}, \\
p(x, y, t) &= (x^2 + y^2 - 1)^2e^{-t}, \\
g_1(x, y, t) &= -\frac{1}{4}ye^{-t} - \frac{1}{16}x(1-x^2)e^{-2t} + 4(x^2 + y^2 - 1)e^{-t}x, \\
g_2(x, y, t) &= \frac{1}{4}x(1-x^2)e^{-t} + \frac{1}{4}ye^{-t}\left(-\frac{1}{4}(1-x^2)e^{-t} + \frac{1}{2}x^2e^{-t}\right) - \frac{3}{2}xe^{-t} + 4(x^2 + y^2 - 1)e^{-t}y.
\end{aligned}$$

Here we have

$$\begin{aligned}
\frac{\partial u}{\partial x} &= 0, & \frac{\partial u}{\partial y} &= \frac{1}{4}e^{-t}, \\
\frac{\partial^2 u}{\partial x^2} &= 0, & \frac{\partial^2 u}{\partial y^2} &= 0, \\
\frac{\partial v}{\partial x} &= \frac{1}{4}e^{-t}(-1 + 3x^2), & \frac{\partial v}{\partial y} &= 0, \\
\frac{\partial^2 v}{\partial x^2} &= \frac{3}{2}xe^{-t}, & \frac{\partial^2 v}{\partial y^2} &= 0, \\
\frac{\partial p}{\partial x} &= 4(x^2 + y^2 - 1)e^{-t}x, & \frac{\partial p}{\partial y} &= 4(x^2 + y^2 - 1)e^{-t}y.
\end{aligned}$$

N	$\ u_{\text{error}}\ _2$	Ratio	$\ v_{\text{error}}\ _2$	Ratio	$\ p_{\text{error}}\ _2$	Ratio
16	5.3838E-04	-	8.1989E-04	-	1.3604E-02	-
32	1.4644E-04	3.6764	2.2699E-04	3.6120	4.3058E-03	3.1594
64	3.8076E-05	3.8459	5.9219E-05	3.8330	1.4074E-03	3.0594
128	9.6009E-06	3.9658	1.4984E-05	3.9521	3.8226E-04	3.6817

Table 10: A grid refinement analysis in L^2 -Norm for Example(2).

N	$\ u_{\text{error}}\ _\infty$	Ratio	$\ v_{\text{error}}\ _\infty$	Ratio	$\ p_{\text{error}}\ _\infty$	Ratio
16	1.3922E-03	-	3.2717E-03	-	8.6817E-02	-
32	4.0255E-04	3.4584	9.6117E-04	3.4038	4.8699E-02	1.7827
64	1.1010E-04	3.6562	2.5788E-04	3.7271	2.6233E-02	1.8564
128	2.8170E-05	3.9084	6.6552E-05	3.8748	1.3360E-02	1.9635

Table 11: A grid refinement analysis in L^∞ -Norm for Example(2).

4.3.2 Cavity Flow

Example (3). Let $\Omega = [0, 1] \times [0, 1]$ be computational domain. Assume that flow is driven by the upper wall. That is, we have the boundary condition

$$\mathbf{u} = \mathbf{0} \text{ on } \partial\Omega \setminus \partial\Omega_{\text{lid}}$$

and

$$\mathbf{u} = (u, v) = (1, 0) \text{ on } \Omega_{\text{lid}},$$

where $\Omega_{\text{lid}} = \{(x, y) \in \partial\Omega, y = 1\}$. Let $N = 128$ be the number of partition. We compare our result with [14].

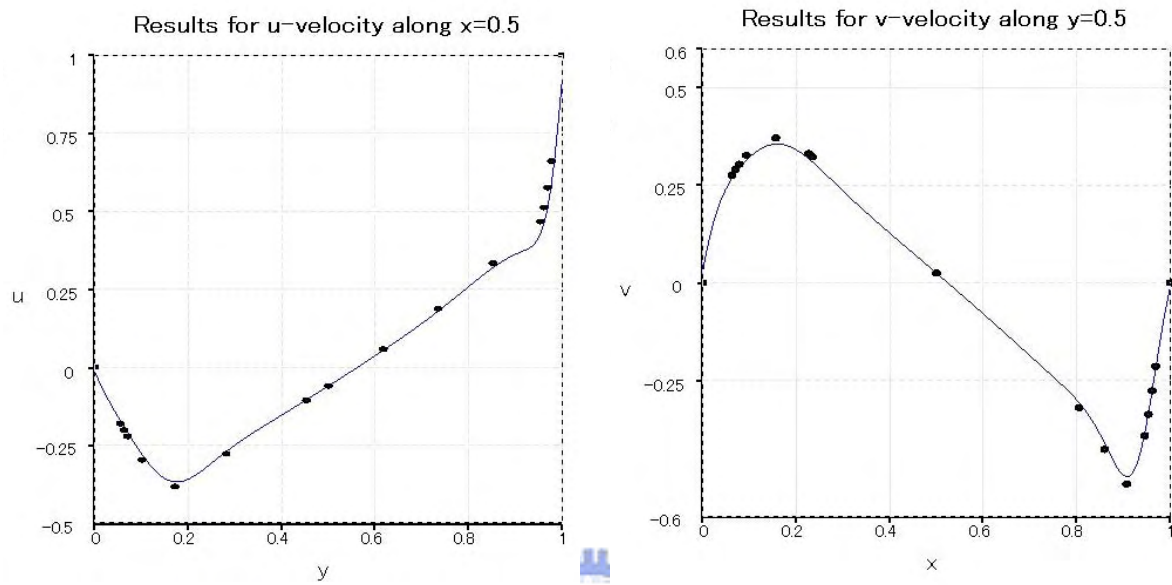


Figure 7: Figures of velocities u and v on $x = 0.5$ and $y = 0.5$ respectively. $Re = 1000$.

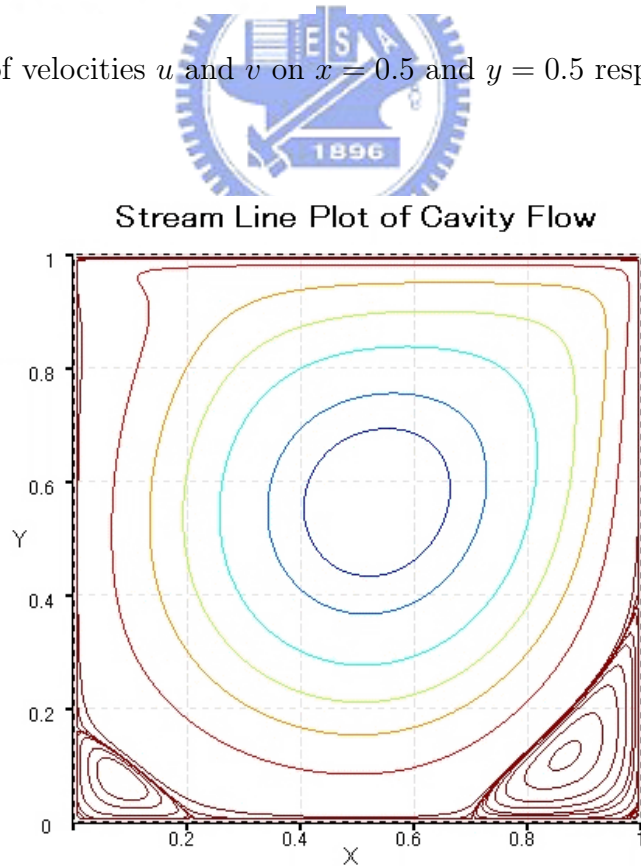


Figure 8: A contour plot of a driven cavity with $Re = 1000$.

4.3.3 Circular Flow with a Line Force

Example (4). Let us assume the curve Γ to be a circle $x^2 + y^2 = \frac{1}{4}$ within a computational domain $\Omega = [-1, 1] \times [-1, 1]$. That is, the interface can be represented by

$$(X(s), Y(s)) = (0.5 \cos \theta, 0.5 \sin \theta)$$

where $\theta = 2s$, $0 \leq s \leq \pi$. Let the pressure and velocity components be

$$u(x, y, t) = \begin{cases} h(t) \left(\frac{y}{r} - 2y \right) & r > \frac{1}{2} \\ 0 & r \leq \frac{1}{2} \end{cases}$$

$$v(x, y, t) = \begin{cases} h(t) \left(-\frac{x}{r} + 2x \right) & r > \frac{1}{2} \\ 0 & r \leq \frac{1}{2} \end{cases}$$

$$p(x, y, t) = \begin{cases} 1 & r > \frac{1}{2} \\ 0 & r \leq \frac{1}{2} \end{cases}$$

where $r = \sqrt{x^2 + y^2}$. It is easy to verify the velocity satisfies the incompressibility constraint, and it is continuous but has finite jump across the interface. That is,

$$\left[\frac{\partial u}{\partial n} \right] = -2h(t) \sin \theta, \quad \left[\frac{\partial v}{\partial n} \right] = 2h(t) \cos \theta, \quad [p] = 1$$

and then

$$F_{\mathbf{n}} = 1, \quad F_{\boldsymbol{\tau}} = -2h(t).$$

The external force \mathbf{g} can be computed to satisfy the Navier-Stokes equation, and yields

$$g_1(x, y, t) = \begin{cases} h(t)^2 \cos \theta \left(4 - 4r - \frac{1}{r} \right) + h(t) \sin \theta \frac{\mu}{r^2} + h'(t) \sin \theta (1 - 2r) & r > \frac{1}{2} \\ 0 & r \leq \frac{1}{2} \end{cases}$$

$$g_2(x, y, t) = \begin{cases} h(t)^2 \sin \theta \left(4 - 4r - \frac{1}{r} \right) - h(t) \cos \theta \frac{\mu}{r^2} - h'(t) \cos \theta (1 - 2r) & r > \frac{1}{2} \\ 0 & r \leq \frac{1}{2} \end{cases}.$$

One can verify that on Γ , ($r = 1/2$),

$$[\mathbf{g}] \cdot \mathbf{n} = h(t)^2 \left(4 - 4r - \frac{1}{r} \right) - 0 = 0.$$

Some results are given here. In our tests, we assume the boundary force is given. We observe that the accuracy in u , v is nearly second order.

N	$\ u\ _\infty$	Ratio	$\ v\ _\infty$	Ratio	$\ p\ _\infty$	Ratio
16	6.2133E-03	0.0000	6.0783E-03	0.0000	1.3309E-01	0.0000
32	2.0049E-03	3.0990	1.9912E-03	3.0526	6.5489E-02	2.0322
64	5.6914E-04	3.5228	5.6695E-04	3.5121	4.4361E-02	1.4762
128	1.4834E-04	3.8367	1.4805E-04	3.8295	1.5033E-02	2.9509
N	$\ u\ _{L^2}$	Ratio	$\ v\ _{L^2}$	Ratio	$\ p\ _{L^2}$	Ratio
16	1.4411E-03	0.0000	1.4221E-03	0.0000	2.5676E-02	0.0000
32	3.8649E-04	3.7287	3.8360E-04	3.7073	1.0012E-02	2.5645
64	1.1245E-04	3.4370	1.1209E-04	3.4223	3.3265E-03	3.0099
128	3.2281E-05	3.4835	3.2235E-05	3.4772	1.2290E-03	2.7066

Table 12: The grid refinement analysis for u , v , and p . $h(t) = 1$.

N	$\ u\ _\infty$	Ratio	$\ v\ _\infty$	Ratio	$\ p\ _\infty$	Ratio
16	1.8770E-03	0.0000	1.8741E-03	0.0000	3.7830E-02	0.0000
32	6.3666E-04	2.9483	6.4121E-04	2.9227	2.0364E-02	1.8577
64	1.8387E-04	3.4625	1.8356E-04	3.4932	1.4330E-02	1.4211
128	4.8301E-05	3.8068	4.8899E-05	3.7539	4.7389E-03	3.0238
N	$\ u\ _{L^2}$	ratio	$\ v\ _{L^2}$	ratio	$\ p\ _{L^2}$	ratio
16	4.3742E-04	0.0000	4.3582E-04	0.0000	7.8480E-03	0.0000
32	1.2249E-04	3.5712	1.2222E-04	3.5660	3.2169E-03	2.4396
64	3.5968E-05	3.4054	3.5932E-05	3.4013	1.0809E-03	2.9762
128	1.0295E-05	3.4937	1.0291E-05	3.4917	4.1351E-04	2.6139

Table 13: The grid refinement analysis for u , v , and p . $h(t) = 1 - e^{-t}$.

5 Concluding Remarks

We have introduced a numerical technique for the computation of interface problems. The result shows the behavior near the immersed interface can be captured well, and in general second order accuracy can be achieved. Moreover, the method is efficient since fast solvers can still be applied.

One of the future works is to solve piecewise constant coefficient Poisson equation

$$\nabla \cdot (\beta_{\pm} \nabla \psi_{\pm}) = \omega_{\pm} \text{ on } \Omega_{\pm}.$$

The immersed interface method can also handle this type of equation. Thus, we can extend this solver to solve piecewise constant coefficient Stokes equation or Navier-Stokes equation.

It also need to be considered that how the spline we constructed approximates the actual interface, and how accurate the curvature and the distance to the foot of perpendicular we obtain. The error generated by approximation of an interface by an spline may lead to the fluctuation of convergent ratio in grid refinement analysis.

It is a challenge that extending this method to three-dimension. Although there are some fast Poisson solvers in 3-D, constructing the interface of surface and finding foot of perpendicular seem to be a heavy task. It is important that maintaining the efficiency and accuracy when extending this method into 3-D.



References

- [1] Alfio Quarteroni, Riccardo Sacco, Fausto Saleri, “Numerical Mathematics.” Texts in Applied Mathematics 37, Springer.
- [2] Alexandre J. Chorin, Jerrold E. Marsden, “A Mathematical Introduction to Fluid Mechanics, 3rd. ed.” Texts in Applied Mathematics 4, Springer-Verlag.
- [3] J. Stoer, R. Bulirsch: “Introduction to Numerical Analysis.” Springer-Verlag, 1983
- [4] Bu-Qing Su, Ding-Yuan Liu: “Computational Geometry: Curve and Surface Modeling.” Academic Press, INC.
- [5] M. C. Lai, “Jump conditions and test examples for the 2D Navier-Stokes equations involving an immersed boundary.”
- [6] R. Cortez, “The Method of Regularized Stokeslets.” *SIAM Journal of Scientific Computing*, vol. 23, No. 4, pp. 1204-1225.
- [7] J. Thomas Beale and Anita T. Layton, “On The Accuracy of Finite Difference Methods for Elliptic Problems with Interfaces.” Aug 19, 2005.
- [8] Zhilin Li and M. C. Lai, “The Immersed Interface Method for the Navier-Stokes Equations with Singular Forces.” *Journal of Computational Physics*, Vol. 171, n. 2, pp. 822-842.
- [9] A. McKenny and L. Greengard and A. Mayo, “A Fast Poisson Solver for Complex Geometries.” *Journal of Computational Physics*, Vol. 118, pp. 348-355.
- [10] M. C. Lai and C. S. Peskin, “An immersed boundary method with formal second-order accuracy and reduced numerical viscosity.” *Journal of Computational Physics*, Vol. 160, pp. 705-719. (2000)
- [11] Boyce E. Griffith, Charles S. Peskin, “On the order of accuracy of the immersed boundary method: Higher order convergence rates for sufficiently smooth problems.” *Journal of Computational Physics*, Vol. 208, pp. 75-105.
- [12] D. Russell. Z. Jane Wang, “A cartesian grid method for modeling multiple moving objects in 2D incompressible viscous flow.” *Journal of Computational Physics* Vol. 191 (2003), pp.177-205.
- [13] Sheng Xu, Z. Jane Wang, “An immersed interface method for simulating the interaction of a fluid with moving boundaries.” *Journal of Computational Physics*, in press.
- [14] U. Ghia, K. N. Ghia, and C. T. Shin, “High-Re Solutions for Incompressible Flow Using the Navier-Stokes Equations and a Multigrid Method.” *Journal of Computational Physics* Vol. 48, pp.387-411.