

**NUMERICAL SIMULATION OF TWO-PHASE FLOW USING THE
LEVEL SET APPROACH**

A Dissertation
Presented to
the Faculty of the Department of Mathematics
University of Houston

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

By
Lan Zhang
December 2014

**NUMERICAL SIMULATION OF TWO-PHASE FLOW USING THE
LEVEL SET APPROACH**

Lan Zhang

APPROVED:

Prof.Suncica Canic
Chairman

Prof.Dmitri Kuzmin
Co-Chairman

Prof.Yuliya Gorb

Prof.Mikhail Perepelitsa

Dean, College of Natural Sciences and Mathematics

NUMERICAL SIMULATION OF TWO-PHASE FLOW USING THE
LEVEL SET APPROACH

An Abstract of a Dissertation
Presented to
the Faculty of the Department of Mathematics
University of Houston

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

By
Lan Zhang
December 2014

Abstract

In this dissertation, we present the numerical schemes in simulating immiscible two-phase flow problems. The goal of this work is to find a unified solution for numerical modeling of any kind of immiscible two-phase flow with a moving interface. The fluid flows are modeled using the Navier Stokes equations with discontinuous coefficients. We introduce a Volume Fraction method to evaluate discontinuous integrals arise from the variational formulations in the Finite Element method. The Volume Fraction method avoids the approximation of the Dirac Delta function, and therefore no regularization procedures are needed. Several operator splitting variants are studied in detail with their linearized and non-linearized versions. The interface is captured using the Level Set approach, where a transport equation is solved numerically with fourth-order scheme without any stabilizations. The surface tension effect is implemented in a semi-implicit way, thus larger time steps can be used compared with the explicit method. A recent, well-developed re-initialization technique is included as a way to preserve the signed distance property of the Level Set function. All mentioned numerical methods are used to build the two-dimensional solvers. Solvers have been tested in single-phase and two-phase flow benchmark problems. In particular, the bubble dynamics are presented to validate the stated numerical schemes.

Contents

1	Introduction	1
1.1	Background and motivation	1
1.2	Dissertation outline	3
2	Mathematical modeling of two-phase flow problem	5
2.1	Flow model	5
2.1.1	Description of motion	5
2.1.2	Conservation of mass and momentum	6
2.1.3	Navier Stokes equations	8
2.1.4	Boundary and initial conditions	10
2.2	Description of the interface	12
2.2.1	Interface representation	12
2.2.2	Level Set method	14
2.2.3	Interfacial conditions	16
2.3	Modeling of two-phase problem	17
2.3.1	Similarity	17
2.3.2	Statement of problem	19
3	Numerical Treatment	21
3.1	Finite Element method (FEM)	23
3.1.1	Fundamentals of FEM	23

3.1.2	Matrix assembly	32
3.2	Numerical solution of the Navier Stokes equations	34
3.2.1	Choice of interpolation functions	34
3.2.2	Time discretization: fractional step projection scheme	36
3.2.3	Numerical solutions for the subproblems	40
3.3	Numerical integration of discontinuous functions	63
3.3.1	Volume Fraction method	66
3.3.2	Numerical test cases and error analysis of Volume Fraction method	69
3.4	Numerical representation of the interface	72
3.4.1	Approximation of Level Set function	72
3.4.2	Approximation of surface tension force	77
3.4.3	Interface re-initialization	83
3.5	Summary of algorithm	87
4	Results	88
4.1	Advection diffusion equation	88
4.2	Flow around cylinder	90
4.3	Driven cavity	99
4.4	Static bubble	114
4.5	Oscillating bubble	119
4.6	Rising bubble	123
5	Conclusions	153
	Bibliography	155

Chapter 1

Introduction

1.1 Background and motivation

A large number of fluid flows encountered in real life and industry are a mixture of phases. These problems are traditionally named as multiphases flows. Multiphase flows could capture flows with different states or phases, namely solid, liquid, or gas. They could also refer to materials with different chemical properties but in the same state or phase. Two-phase flow is a special case of such problems. The liquid liquid system such as oil droplets in water is an example of two-phase flow.

Modeling two-phase flow problem is not an easy task. Empirical correlations have been used for many years. They are based on measurements consisting mostly of low pressure, small diameter data. Extrapolations of correlations to field conditions may result in errors. Lockhart and Martinelli [30], and Beggs and Brill [3] attributed to these methods. Mechanistic models are developed based on the fundamental mechanisms of two-phase flow. Navier Stokes equations are used to represent conservation of momentum and continuity.

Chisholm [5] predicted flow patterns in tube bundles for gas bubble in liquid. OLGA, and LedaFlow are two established commercial tools to simulate the complex flow patterns used in the oil and gas industry currently.

Numerically, some commercial softwares, for example ANSYS Fluent and CFX, have excelled in solving some one-phase Navier Stokes benchmark problems, while extra user modifications and interventions are often necessary when simulating two-phase problems. Moreover, computation efficiency and accuracy are still open problems for those in the Computational Fluid Dynamics (CFD) research community. NASA released their vision for all aspects of CFD simulations [32]. They provided a 15-year road map for CFD development. Hence, it worthy of studying two-phase flow problem using CFD approach. Intuitively, there are some numerical difficulties in simulating immiscible two-phase flows, including:

- Robust solvers are needed for Navier Stokes systems with discontinuous density and viscosity;
- Interface capture and topological changes of interface with capillary force effect;
- Coupling with fluid flows and interface.

With all those numerical difficulties in mind, we present numerical methods for numerical simulations of two-dimensional incompressible two-phase flow problems. The interface is captured using Level Set method. Flows and interface are both based on Finite Element discretization spatially on structured meshes. Operator splitting schemes are applied for time integration. Corresponding detailed subproblems are investigated in detail due to the presence of discontinuous density and viscosity coefficients. Two-phase flow solvers are

built based on mentioned numerical schemes. They have been tested from the one-phase fluid flow governing by Navier Stokes equations, and extended to two-phase problem solver using proposed Volume Fraction method. A recently developed reinitialization technique is also implemented in solvers.

1.2 Dissertation outline

This dissertation is organized as follows: Chapter 2 carefully derives mathematical modeling of fluid flows in a classic manner, and introduces Level Set method to model the interface between two immiscible fluid flows. It then couples the modelings of fluid flows and the interface. Chapter 3 devotes the main computational strategies of two-phase flow problem. It begins with a short review of the Finite Element method. It continues with the detailed numerical discretization techniques to solve the Navier Stokes equations. It follows with a new approach to evaluate the discontinuous coefficients arise from the Navier Stokes equations, named the Volume Fraction method. Numerical test cases and error analysis using the Volume Fraction method to approximate discontinuous integrals are presented as well. This is followed with a section to describe the Finite Element approach to implement Level Set method. The semi-implicit integration method is applied to surface tension force. It is a efficient method because the time step criterion has been removed. The algorithm is completed with a recent developed reinitialization approach, where the interface is preserved by minimizing a cost functional with a penalty term. This method ends up with a nonlinear elliptic problem, and it can be solved using a simple fixed point iteration method. The dissertation is completed with benchmark test results and numerical validations in Chapter 4. Every single part of the operator splitting scheme is verified

using well-developed classical problems. Beginning with Burgers' equation where convection and diffusion terms are considered without pressure and divergence free part, the steady state of single-phase flow around cylinder problem is tested using operator splitting scheme. It follows with the unsteady driven cavity problem. Single-phase flow and flows with different densities and viscosities are presented. Surface tension force validation is carried out using static bubble and oscillating bubble tests. Solvers are finally tested using rising bubble test. Results and measured quantities are compared with reference data from other research groups. Solutions are also presented using commercial software for plausible options. The dissertation is concluded with all current work in simulating two-phase flow problems.

Chapter 2

Mathematical modeling of two-phase flow problem

An important consideration to simulate the fluid flow problems numerically is to understand the meaning of each equation. Classical mathematical description of fluid mechanics is derived from continuum mechanics.

2.1 Flow model

2.1.1 Description of motion

In classic fluid mechanics, the description of motion can be stated in differential form, applicable at an infinitesimal control volume, or it can also be stated in the integral form, applicable at a control volume of a finite size. No matter which form one uses, they all lead to the same equations. We use the integral form in this dissertation to derive all the

motions mathematically. In classic fluid mechanics, there are two ways to describe the flow field: the Lagrangian description and the Eulerian description. In the Lagrangian description, one follows an individual particle as it moves through time and space. In the Eulerian description, one focus on a specific location in space where the fluid flows pass by. The Lagrangian descrtiption is related to Eulerian description of motion via the following substantial (or material) derivatives:

Definition 2.1. The *substantial derivative* of some physical quantity describes the time rate of change for a moving fluid particle. Let $F(\vec{x}, t)$ be any physical quantity, and \vec{u} be the velocity of fluid flow. The substantial derivative of F is defined to be

$$\frac{dF}{dt} = \frac{\partial F}{\partial t} + \vec{u} \cdot \nabla F \quad (2.1)$$

where the substantial derivative can be expressed as the sum of local derivative and the convective derivative. This substantial derivative will be essentially useful when the Eulerian description of motion is adopted.

2.1.2 Conservation of mass and momentum

All fluid mechanics is based on the conservation laws for mass, momentum, and energy. Conservation of energy is based on first law of thermodynamics. We do not consider this aspect in our work. We focus on conservation of mass and conservation of momentum only. As mentioned in previous section, we are going to derive conservation of mass and momentum using an Eulerian description in integral form. The following results will be helpful:

Theorem 2.2. *For any fluid property $F(\vec{x}, t)$, the time derivative of an integral of $F(\vec{x}, t)$*

over a fixed control volume Ω is

$$\frac{d}{dt} \int_{\Omega} F(\vec{x}, t) d\Omega = \int_{\Omega} \frac{\partial F(\vec{x}, t)}{\partial t} d\Omega \quad (2.2)$$

and the time derivative of an integral of $F(\vec{x}, t)$ over a moving control volume $\Omega(t)$ is

$$\frac{d}{dt} \int_{\Omega(t)} F(\vec{x}, t) d\Omega = \int_{\Omega(t)=\Omega} \frac{\partial F(\vec{x}, t)}{\partial t} d\Omega + \int_{S(t)=S} F(\vec{x}, t) \vec{u} \cdot \vec{n} dS \quad (2.3)$$

Equation (2.3) is also named Reynolds transport theorem, which states the rate of change in a moving volume is the sum of rate of change in a fixed volume and the convective transfer through the surface. With all these preparations, we begin with conservation of mass using a fixed control volume.

Conservation of mass principle says the rate of increase of mass within any fixed volume must be equal to the rate of inflow through the boundary. Mathematically it means:

$$\int_{\Omega} \frac{\partial \rho}{\partial t} d\Omega = - \int_S \rho \vec{u} \cdot \vec{n} dS \quad (2.4)$$

Apply divergence theorem to the right hand side, then

$$\int_{\Omega} \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{u}) d\Omega = 0 \quad (2.5)$$

Notice this (2.5) holds for any fixed volume, thus we come up with the following **conservation of mass** equation:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{u}) = 0 \quad (2.6)$$

Conservation of momentum can be derived using a moving volume based on Newton's

second law, which says the rate of change of momentum equals the sum of body force throughout the volume and the surface force at the surface. Mathematically it means:

$$\frac{d}{dt} \int_{\Omega(t)} \rho \vec{u} d\Omega = \int_{\Omega(t)} \rho \vec{g} d\Omega + \int_S \sigma \cdot \vec{n} dS \quad (2.7)$$

where \vec{g} is the gravitational acceleration, $\sigma = -pI + \tau$ is the stress tensor, p is pressure, I is identity matrix, and τ is viscous stress. Apply Reynolds transport theorem as well as divergence theorem,

$$\int_{\Omega} \frac{\partial(\rho \vec{u})}{\partial t} + \nabla \cdot (\rho \vec{u} \otimes \vec{u}) d\Omega = \int_{\Omega} \rho \vec{g} + \nabla \cdot \sigma d\Omega \quad (2.8)$$

Again, (2.8) holds for any volume, thus we come up with the following **conservation of momentum** equation:

$$\frac{\partial(\rho \vec{u})}{\partial t} + \nabla \cdot (\rho \vec{u} \otimes \vec{u}) = \rho \vec{g} + \nabla \cdot (-pI + \tau) \quad (2.9)$$

2.1.3 Navier Stokes equations

Under the assumption that the fluid flow is homogeneous, meaning densities and viscosities are constant within the fluid, and it is incompressible, meaning the rate of change of the density with respect to time is zero, i.e. $\frac{\partial \rho}{\partial t} = 0$, the conservation of mass (2.6) reduce to

$$\nabla \cdot \vec{u} = 0 \quad (2.10)$$

Besides, as we consider viscous Newtonian fluid, shear stress is assumed to be proportional to velocity gradient, i.e. $\tau = \mu(\nabla \vec{u} + \nabla \vec{u}^T)$. Rewrite the second term in conservation of momentum (2.9), using the identity $\nabla \cdot (\rho \vec{u} \otimes \vec{u}) = \vec{u} \cdot \nabla(\rho \vec{u}) + \rho \vec{u}(\nabla \cdot \vec{u})$, and conservation

of mass, the second term reduce to $\nabla \cdot (\rho \vec{u} \otimes \vec{u}) = \rho \vec{u} \cdot \nabla \vec{u}$, which leads to the conservation of momentum for viscous Newtonian fluid

$$\rho \left(\frac{\partial \vec{u}}{\partial t} + (\vec{u} \cdot \nabla) \vec{u} \right) = \rho \vec{g} - \nabla p + \nabla \cdot (\mu (\nabla \vec{u} + \nabla \vec{u}^T)) \quad (2.11)$$

Together with the incompressibility condition (2.10), the following systems describe incompressible viscous Newtonian fluid flow and are named Navier Stokes equations:

$$\begin{aligned} \rho \left(\frac{\partial \vec{u}}{\partial t} + (\vec{u} \cdot \nabla) \vec{u} \right) &= \rho \vec{g} - \nabla p + \nabla \cdot (\mu (\nabla \vec{u} + \nabla \vec{u}^T)) \\ \nabla \cdot \vec{u} &= 0 \end{aligned} \quad (2.12)$$

The system must be completed with corresponding initial and boundary conditions, which will be discussed in later subsections.

Remark 2.3. *If μ is constant, namely the case of single-phase flow, the term $\mu(\nabla \vec{u} + \nabla \vec{u}^T)$ simplifies to*

$$\nabla \cdot (\mu (\nabla \vec{u} + \nabla \vec{u}^T)) = \mu \nabla \cdot (\nabla \vec{u} + \nabla \vec{u}^T) = \mu (\Delta \vec{u})$$

since $\nabla \cdot \vec{u} = 0$.

Navier Stokes equations are widely used to describing unsteady, incompressible, viscous Newtonian single-phase flows. We still want to use Navier Stokes equations to model two-phase immiscible fluid flows by assuming within each fluid, it is unsteady, incompressible, viscous Newtonian homogeneous fluid flow, but densities and viscosities are different for two different flows. There are essentially two approaches to realize this goal. One way is to model each fluid flow separately using its own Navier Stokes equations and corresponding

initial and boundary conditions. It is computationally expensive because we have to solve two Navier Stokes equations, and it is difficult to capture the interaction between two fluids because two Navier Stokes equations are related to each other. Another way is to model the two-phase flow using the same Navier Stokes equations with densities and viscosities as functions of space. By doing so, only one system need to be solved, and in this case the interaction between two fluids can be captured as an additional boundary condition for the whole system. Besides, we will see in section (3.3) that dealing the discontinuous coefficients is very handy when using the Volume Fraction approach. In summary, the two immiscible fluid flows can captured by the following system:

$$\rho(\vec{x})\left(\frac{\partial \vec{u}}{\partial t} + (\vec{u} \cdot \nabla)\vec{u}\right) = \rho(\vec{x})\vec{g} - \nabla p + \nabla \cdot (\mu(\vec{x})(\nabla \vec{u} + \nabla \vec{u}^T)) \quad (2.13)$$

$$\nabla \cdot \vec{u} = 0$$

We do not consider phase transition and mass transfer here. So additional conditions should hold on the interface, namely velocity must be assumed to be continuous across the interface. Besides, surface tension force as the interfacial force should be incorporated. Details of analysis will be presented in the description of interface section. In addition, suitable boundary and initial conditions have to be added, which are stated in the following subsection.

2.1.4 Boundary and initial conditions

In applications, we can always restrict ourselves to some domain of interest. This can be simplify our problem from computational point of view. Boundary and initial conditions are essentially important in this artificial domain. Theoretically, we do need those conditions to insure the problem is well-posed, meaning a solution exists, the solution is unique, and it

changes continuously with the input. Typically, there are two types of boundary conditions for the velocity field: Dirichlet boundary conditions, and Neumann boundary conditions. Dirichlet boundary conditions specify the value of velocity on the boundary:

$$\vec{u}(\vec{x}, t) = \vec{b} \quad \text{on} \quad \Sigma_D$$

where Σ_D is the Dirichlet part of boundary in domain Ω . This type of boundary condition is also known as the non-slip condition when the boundary is a solid wall contacting the fluid, i.e. the velocity boundary value \vec{b} is equal to the velocity of the wall. And when we consider a computational domain where the fluid enters the domain, the Dirichlet type of boundary condition for velocity and pressure fields must be specified at this inflow part of boundary.

The Neumann boundary condition describes the flux of the fluid,

$$\vec{u}(\vec{x}, t) \cdot \vec{n} = b \quad \text{on} \quad \Sigma_N$$

Here Σ_N is the Neumann part of boundary in domain Ω , and \vec{n} is the outward normal of the boundary as usual. This type of boundary conditions can be specified when the outflow part is included in the computational domain. In this case, zero normal stress and zero tangential velocity on the outflow part of boundary should be specified.

Dirichlet boundary conditions and Neumann boundary conditions describe different physics phenomena, and from a computational point of view, it is very important to distinguish them in the Finite Element regime. The Dirichlet boundary condition is named principle boundary condition in standard the Galerkin Finite Element approach, and all

test functions have to be zero on the Dirichlet boundary, whereas there are no such restrictions on the boundary of the Neumann type. In another Finite Element approach, like the Mixed Finite Element method, the boundary condition of the Neumann type is the principle boundary condition and the Dirichlet boundary condition is a natural one.

The following initial condition for the velocity field must be given in the computational domain as well:

$$\vec{u}(\vec{x}, 0) = \vec{u}^{(0)} \quad \text{in } \Omega$$

and it has to be divergence free due to the incompressibility condition, namely,

$$\nabla \cdot \vec{u}^{(0)} = 0 \quad \text{in } \Omega$$

Notice we do not have to specify the initial condition for the pressure field at this point, because there is no time derivative with respect to the pressure term in the system. However, when Dirichlet boundary conditions are prescribed everywhere in the computational domain, pressure is determined only up to an arbitrary constant due to the gradient term in the Navier Stokes equations. In this case we do need to impose the value of pressure at one point to make the solution uniquely defined.

2.2 Description of the interface

2.2.1 Interface representation

In two-phase flow simulation, it is essential to capture the position and shape of the interface separating two immiscible flows. In general, there are two approaches to track the motion of interface: explicit and implicit. When the interface is represented explicitly,

parameterization of the interface is needed. The coordinates of the interface are explicitly traced using marker particles. The Front-tracking method [58], Marker in Cell (MAC) method [64], and Immersed Boundary method [40] fall into this category. Since a large number of particles can be used, the interface can be localized to good accuracy. However, lots of numerical complexities are introduced as well. Handling break up and merging of interfaces are also very difficult in this scenario. In the second approach, the interface is represented by a function instead, where no explicit representation of the interface is needed. All topological changes of the interface are represented by the continuous evolution of the corresponding function defined the interface. The Volume of Fluid method (VOF) is an example of implicit method, which was introduced by Hirt and Nichols [36]. Based on Noh and Woodward's work of Simple Line Interface Calculation (SLIC) [65], the basic idea of the VOF method is that given the interface on a computational domain with fixed grid, functions named "cell fractions" are assigned on each grid cell such that they represent the fraction of the cell containing fluids inside the interface. The interface locations and evolutions of interface are based on reconstructing and updating such functions. The VOF method is attractive due to its mass conservative property. Level Set method is another example of implicit method. It was introduced by Osher and Sethian [50], and is also based on a fixed grid. The interface is defined as a zero Level Set of a continuous function on the whole computational domain, and the evolution of the interface can be obtained by updating the Level Set function. The interface normal and curvature approximations are straightforward using the level set function. Besides, due to the Eulerian natural of implicit techniques, the Level Set method is very powerful in handling topological changes of interfaces. In the following section, we focus on the Level Set method.

2.2.2 Level Set method

The basic idea of the Level Set method is to define the interface Γ as the zero level set of a higher dimensional function ϕ , that is defined initially as,

$$\phi(\vec{x}, t = 0) = \begin{cases} \text{dist}(\Gamma, \vec{x}) & \vec{x} \in \Omega_1 \\ 0 & \vec{x} \in \Gamma \\ -\text{dist}(\Gamma, \vec{x}) & \vec{x} \in \Omega_2 \end{cases}$$

where $\text{dist}(\Gamma, \vec{x})$ is the shortest distance between \vec{x} to Γ . Regions that two immiscible fluids occupy, say, Ω_1 and Ω_2 , can be determined by different signs of the Level Set function respectively. For example, a zero level of $\phi(\vec{x}) = 0.25 - \sqrt{(x - 0.5)^2 + (y - 0.5)^2}$ defines a circle of radius 0.25 centered at (0.5, 0.5). In the following figures, the solid black lines denotes the zero Level Set.

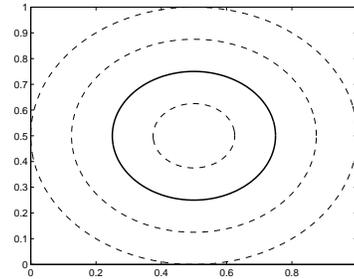
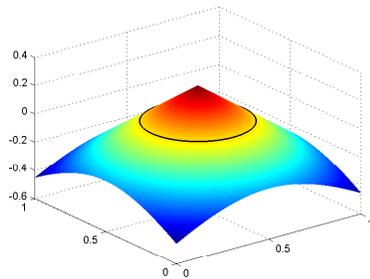


Figure 2.1: (a) the Level Set function Figure 2.2: (b) contours of Level Set function

Besides, the normal vector \vec{n} and the curvature κ can be defined globally as

$$\vec{n} = \frac{\nabla \phi}{|\nabla \phi|} \tag{2.14}$$

and

$$\kappa = -\nabla \cdot \vec{n} \quad (2.15)$$

Recognized that at a later time $t > 0$, the following must hold for the moving interface

$$\phi(\vec{x}(t), t) = 0$$

To evolve this Level Set function, differentiate with respect to time t to yield,

$$\frac{\partial \phi}{\partial t} + \nabla \phi \cdot \frac{\partial \vec{x}}{\partial t} = 0$$

Since a point on the interface should move according to the velocity field, we have $\frac{\partial \vec{x}}{\partial t} = \vec{u}$.

Thus the following equation must hold globally for the interface

$$\frac{\partial \phi}{\partial t} + \vec{u} \cdot \nabla \phi = 0 \quad (2.16)$$

We derived an advection equation for the Level Set function with initial condition $\phi(\vec{x}, t = 0) = \phi^{(0)}$. Another thing to mention is that the boundary condition should be given only on the inlet of the boundary domain. Besides, the Level Set function may not preserve its nice property namely, a signed distance function as it transported with the velocity field. Especially in two-phase simulation, this is critical important since an acceptable conservation of mass is needed for each fluids. To recover the signed distance property, a reinitialization step should be included, which will be summarized and analyzed in Chapter 3.

2.2.3 Interfacial conditions

Since the two fluid flows are assumed to be immiscible and incompressible, more conditions on the interface should be considered. We will see in this section that those interfacial conditions are actually extra boundary conditions of two-phase flow modeling. First of all, mass conservation should also hold on both of the fluids. Thus there should be no mass jump across the interface, i.e. the velocity field should be continuous across the interface.

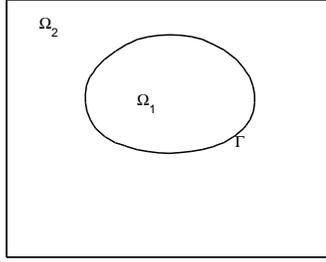


Figure 2.3: Two-phase flow

For example, on the interface Γ in Figure (2.3), the following should hold

$$[\vec{u}]|_{\Gamma} = 0 \tag{2.17}$$

where $[\vec{u}]$ denotes the jump of velocity field \vec{u} on the interface Γ .

In addition, as two immiscible fluid flows are in contact, the intermolecular attractive force brings an extra force, named surface tension force, which makes the interface to contract. So, the following boundary condition should hold on the interface Γ as well

$$-[-pI + \mu(\nabla\vec{u} + \nabla\vec{u}^T)]|_{\Gamma} \cdot \vec{n} = \sigma\kappa\vec{n} \tag{2.18}$$

which states that on the interface the jump in the normal stress should be balanced by the surface tension force. Here σ is the surface tension coefficient, which depends on the two fluids in contact and the temperature. It is assumed to be constant for simplicity. κ is the curvature of interface Γ and \vec{n} is the outward normal vector. Another way to consider this surface tension force is named the Continuum Surface Force (CSF) model, see [68], [38]. It includes the surface tension force in the momentum equation as an extra source term by introducing the Dirac Delta function, namely,

$$\rho(\vec{x})\left(\frac{\partial \vec{u}}{\partial t} + (\vec{u} \cdot \nabla)\vec{u}\right) = \rho(\vec{x})\vec{g} + \vec{f} - \nabla p + \nabla \cdot (\mu(\vec{x})(\nabla \vec{u} + \nabla \vec{u}^T)) \quad (2.19)$$

where

$$\vec{f} = \sigma \kappa \vec{n} \delta(\Gamma, \vec{x})$$

This formulation is physically reasonable. It can be actually derived from conservation of mass and momentum in the domain by exercising Stokes' theorem. However, for the CSF model, the location of interface $\Gamma(t)$ has to be known at each time step. More details of localization of the interface and how to handle the surface tension force will be revisited in Chapter 3.

2.3 Modeling of two-phase problem

2.3.1 Similarity

Similarity is characterized by a set of non-dimensional parameters, in which data are unified. These parameters are very important in model simplification, because they can identify which physics effect dominates the problem and which ones are less relevant and

therefore can be neglected. Non-dimensional parameters are essentially important for designing model tests when full-scale tests can not be performed, such as predicting the flow properties around an aircraft.

When the governing equations are known, non-dimensional parameters can be determined directly from them. In our two-phase model, the analysis of surface tension effects brings additional dimensionless parameters in which surface tension is compared with other effects such as body forces and viscous stress. Introduce characteristic velocity scale U and characteristic length scale l , and accordingly the following non-dimensional variables are:

$$x' = \frac{x}{l}, \quad y' = \frac{y}{l}, \quad t' = \frac{tU}{l}, \quad \vec{u}' = \frac{\vec{u}}{U}, \quad p' = \frac{p}{\rho U^2}$$

Substitute all these variables into the momentum equation (2.19) as well as incompressibility condition,

$$\begin{aligned} \frac{\rho U^2}{l} \left(\frac{\partial \vec{u}'}{\partial t'} + (\vec{u}' \cdot \nabla') \vec{u}' \right) &= \rho \vec{g} + \vec{f} - \frac{\rho U^2}{l} \nabla' p' + \nabla' \cdot \left(\frac{U}{l^2} (\mu (\nabla' \vec{u}' + \nabla' \vec{u}'^T)) \right) \\ \nabla' \cdot \vec{u}' &= 0 \end{aligned}$$

Drop the factor $\frac{\rho U^2}{l}$ and after simplification, we can derive non-dimensional Navier Stokes equations in the following form:

$$\begin{aligned} \frac{\partial \vec{u}'}{\partial t'} + (\vec{u}' \cdot \nabla') \vec{u}' &= \frac{l \vec{g}}{U^2} + \frac{\sigma}{\rho U^2 l} \kappa' \vec{n} \delta(\Gamma, \vec{x}) - \nabla' p' + \nabla' \cdot \left(\frac{\mu}{\rho U l} (\nabla' \vec{u}' + \nabla' \vec{u}'^T) \right) \quad (2.20) \\ \nabla' \cdot \vec{u}' &= 0 \end{aligned}$$

The inverse of coefficient in front of the viscous term has been given special name, the

Reynolds number:

$$Re \equiv \frac{\rho U l}{\mu} \quad (2.21)$$

The Reynolds number is one of the most important non-dimensional parameters. It quantifies the relationship between the inertial and viscous forces. A low Reynolds number, where viscous force is dominant, characterizes laminar flow with smooth fluid motion, and a high Reynolds number, where fluids are dominated by inertial force, characterizes turbulent flow which tend to produce chaotic eddies, vortices, and other flow instabilities.

Another non-dimensional parameters is named the Froude number which is defined to be:

$$Fr \equiv \frac{U^2}{lg} \quad (2.22)$$

It relates the body's inertia to gravitational forces.

Weber number is defined as the inverse of the coefficient in front of the surface tension force, which is:

$$We \equiv \frac{\rho U^2 l}{\sigma} \quad (2.23)$$

It is used to characterize the relation between the inertia and surface tension effects. This quantity is useful in analyzing thin film flows and the formation of droplets and bubbles.

2.3.2 Statement of problem

Under assumptions of immiscibility and incompressibility, our two-phase flow problem is fully captured by the following system of equations:

$$\rho(\vec{x})\left(\frac{\partial \vec{u}}{\partial t} + (\vec{u} \cdot \nabla)\vec{u}\right) - \nabla \cdot (\mu(\vec{x})(\nabla \vec{u} + \nabla \vec{u}^T)) + \nabla p = \rho(\vec{x})\vec{g} \quad \text{in } \Omega \times [0, T] \quad (2.24)$$

$$\nabla \cdot \vec{u} = 0 \quad (2.25)$$

$$[\vec{u}]_\Gamma = \vec{0}, \quad -[-pI + \mu(\vec{x})(\nabla \vec{u} + \nabla \vec{u}^T)]|_\Gamma \cdot \vec{n} = \sigma \kappa \vec{n} \quad (2.26)$$

with appropriate initial and boundary conditions. Here the entire computational domain is $\Omega = \Omega_1 \cup \Omega_2$, and the interface is Γ . $\rho(\vec{x}) = \rho_1$ in Ω_1 and ρ_2 in Ω_2 , and $\mu(\vec{x}) = \mu_1$ in Ω_1 and μ_2 in Ω_2 . In addition, the interface is captured by Level Set method, where the interface is transported with the flow. In the following chapter, numerical schemes for the flow as well as the interface are considered in detail. Both the flows and the interface are approximated using the Finite Element method.

Chapter 3

Numerical Treatment

As we have seen in the previous chapter, our problem brought up three main questions: how to solve flow problems captured by incompressible Navier Stokes equations; how to convect the interface with capillary force; and how to couple flows and the interface through velocity and pressure fields. In this chapter, we answer those three questions step by step, namely solving flow problems using fractional step projection approach with fixed interface, then advecting the interface with a known velocity field, and finally coupling fluid flows with the interface through discontinuous densities/viscosities coefficients and capillary forces.

We begin with the discretization of the Navier Stokes equations. Numerical methods for the solution of incompressible viscous Navier Stokes equations have been studied extensively, and several numerical approaches have been developed even for various formulation of the problem. Nowadays, there are many numerical approaches to solve this nonlinear time-dependent partial differential equations. Classical spatial discretization is based on Finite Difference method, Finite Volume method and Finite Element method. The Finite Difference method and Finite Volume methods are more popular among engineers due to

the easy implementation and conservation of physical properties respectively. However, the Finite Difference method has difficulties when dealing with curved boundaries. Moreover they both do not perform well under stability and convergence analysis. The Finite Element method outperforms the Finite Difference method as well as the Finite Volume method, especially when we have complex geometry and/or boundary conditions. Since Finite Element method has been successfully implemented and validated in numerous literatures over complex domains with high accuracy and stability, we focus on the Finite Element method in this dissertation.

Unfortunately, spatial discretization is not the only difficulties for unsteady problems. An appropriate time discretization technique is an important consideration in addition to spatial stability. Fractional Step (or Operator Splitting) methods are very popular for time integration of unsteady problems. There are many forms of operator splitting schemes; however, they all involve splitting a complex problem into simple ones. In this dissertation, we present fractional step projection methods, which first originated by Chorin [6] [7] and Temam [54], and are widely applied for incompressible Navier Stokes equations. The idea is to compute the velocities by different physical process, which leads to different types of unsteady partial differential equations, and compute the velocity and pressure fields separately due to projection method. Details of computation strategy are presented in section 3.2.

Usually, the first task to solve an unsteady PDE via Finite Element methods is to discretize the equations in space, which leads to an ODE (Ordinary Differential Equation). Then use numerical integration techniques for time to complete the discretization of PDE. Which discretization comes first makes no difference for simple problems. However, as we have pointed out, all the numerical difficulties, for example, unsteady, nonlinearity,

and stability etc., we are going to perform the time discretization first using operator splitting technique. In each subproblem, we follow the typical way, i.e. perform the spatial discretization before the time discretization. We briefly introduce the ideas of the Finite Element method first in the following section.

3.1 Finite Element method (FEM)

3.1.1 Fundamentals of FEM

The Finite Element method (FEM) was first introduced to solve complex elasticity and structural analysis problems in civil and aeronautical engineering, and it has been shown to be a valuable tool for solving Navier Stokes equations in numerous articles. The standard Galerkin Finite Element method [47] is based on the weak formulation, and it is a systematic approach to discretize the PDEs into algebraic systems. Firstly, multiply the residual of PDEs by a test function which vanishes on the Dirichlet boundary, if any, and integrate over the computational domain using integration by parts. Secondly, represent the approximation solution as a linear combination of basis functions defined on a given mesh. Then plug in the approximate solution and the basis function in the weak formulation. The final step is to solve the resulting algebraic system for the nodal values of approximate solution. The Finite Element method encompasses many other methods because it allows an accurate representation of complex geometry. It also passes higher derivatives in the original PDEs onto derivatives of test functions, thus it reduces the regularities of the solution.

3.1.1.1 Weighted residual formulation

In the context of weak formulation, we need to define two classes of functions: the test function and the admissible solutions. Both of test functions and admissible functions are defined on Sobolev space, i.e. functions themselves and their first derivative are squared integrable over the computational domain. However, test functions and admissible functions have different properties on the Dirichlet boundary. For a computational domain Ω , the test functions are defined as follows:

$$\mathcal{V} = \{w \in H^1(\Omega) | w = 0 \text{ on } \Gamma_D\} \quad (3.1)$$

and the admissible solutions are defined on similar space to the test functions, except that these solutions have to satisfy the Dirichlet boundary conditions. The admissible solutions are defined as follows:

$$\mathcal{S} = \{u \in H^1(\Omega) | u = u_D \text{ on } \Gamma_D\} \quad (3.2)$$

Thus, for non-homogeneous Dirichlet boundary conditions, test functions and admissible solutions live in different spaces, while for homogeneous Dirichlet boundary conditions, they all live in:

$$H_0^1(\Omega) \equiv \{v \in H^1(\Omega) | v = 0 \text{ on } \Gamma_D\} \quad (3.3)$$

3.1.1.2 Finite element spatial discretization

Spaces (3.1), (3.2), and (3.3) contain infinitely many functions. In FEM, those spaces are approximated by corresponding finite dimensional subsets of these spaces. The finite

element spaces are based on *triangulation* of the computational domain.

Definition 3.1. The computational domain Ω is subdivided into many subdomains Ω_k called elements. We define \mathcal{T}_h as *triangulation* of the computational domain if

$$\bar{\Omega} = \bigcup_{\Omega_k \in \mathcal{T}_h} \Omega_k$$

$$\Omega_i \cap \Omega_j = \emptyset \quad \text{if } i \neq j$$

The nonempty intersection of any two elements must be a common vertex /edge/face of the mesh only.

The corresponding interpolation spaces for test functions and admissible solutions are defined on the following spaces:

$$\mathcal{V}^h = \{w^h \in H^1(\Omega) | w^h|_K \in \mathcal{P}_m(k) \quad \forall K \in \mathcal{T}_h \quad \text{and} \quad w^h = 0 \quad \text{on} \quad \Gamma_D\} \quad (3.4)$$

$$\mathcal{S}^h = \{u^h \in H^1(\Omega) | u^h|_K \in \mathcal{P}_m(k) \quad \forall K \in \mathcal{T}_h \quad \text{and} \quad u^h = u_D^h \quad \text{on} \quad \Gamma_D\} \quad (3.5)$$

where \mathcal{P}_m is the finite element interpolating space. In this way, any function can be uniquely defined by a finite number of degrees of freedom. To summarize, we define *Finite Element* as follows:

Definition 3.2. The *Finite Element* is a triple (K, \mathcal{P}, Σ) , where

1. K is a closed subset of $\bar{\Omega}$
2. \mathcal{P} is the polynomial space for the basis functions

3. Σ is the set of local degrees of freedom

and the basis functions have the property:

$$\lambda_j(x, y) = \delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases} \quad (3.6)$$

Linear (P1) and quadratic (P2) basis functions in triangular finite element are constructed using two-dimensional barycentric coordinates with the following numbering in each element:

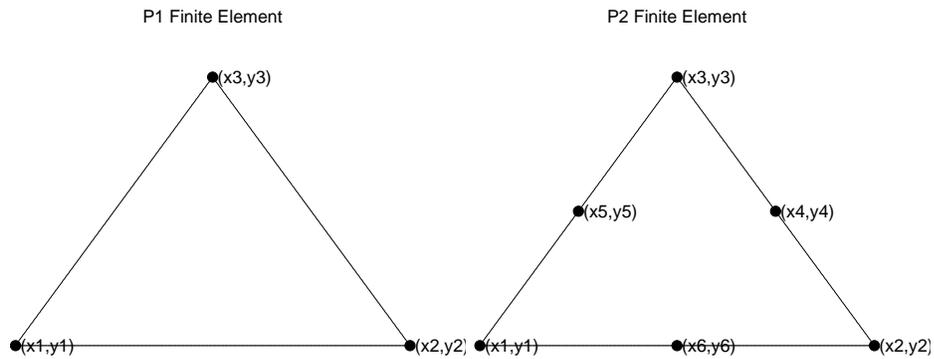


Figure 3.1: Ordering of vertex for P1 and P2 FEM

Denote

$$\det(A_1) = \begin{pmatrix} 1 & x & y \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{pmatrix}, \quad \det(A_2) = \begin{pmatrix} 1 & x_1 & y_1 \\ 1 & x & y \\ 1 & x_3 & y_3 \end{pmatrix}, \quad \det(A_3) = \begin{pmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x & y \end{pmatrix}$$

Notice

$$\begin{pmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{pmatrix} = 2\Delta_e$$

where Δ_e is the area of each element, and P1 basis functions are constructed as follows:

$$\begin{aligned} \lambda_1 &= \frac{\det(A_1)}{2\Delta_e} \\ \lambda_2 &= \frac{\det(A_2)}{2\Delta_e} \\ \lambda_3 &= \frac{\det(A_3)}{2\Delta_e} \end{aligned}$$

The P1 basis function can be expressed in the following way:

$$\begin{aligned} \lambda_1 &= \frac{1}{2\Delta_e}(a_1 + b_1x + c_1y) \\ \lambda_2 &= \frac{1}{2\Delta_e}(a_2 + b_2x + c_2y) \\ \lambda_3 &= \frac{1}{2\Delta_e}(a_3 + b_3x + c_3y) \end{aligned} \tag{3.7}$$

where

$$\begin{aligned} a_1 &= x_2y_3 - x_3y_2, & b_1 &= y_2 - y_3, & c_1 &= x_3 - x_2 \\ a_2 &= x_3y_1 - x_1y_3, & b_2 &= y_3 - y_1, & c_2 &= x_1 - x_3 \\ a_3 &= x_1y_2 - x_2y_1, & b_3 &= y_1 - y_2, & c_3 &= x_2 - x_1 \end{aligned} \tag{3.8}$$

P2 basis functions are constructed by P1 basis functions:

$$\begin{aligned}
\psi_1 &= \lambda_1(2\lambda_1 - 1) \\
\psi_2 &= \lambda_2(2\lambda_2 - 1) \\
\psi_3 &= \lambda_3(2\lambda_3 - 1) \\
\psi_4 &= 4\lambda_2\lambda_3 \\
\psi_5 &= 4\lambda_1\lambda_3 \\
\psi_6 &= 4\lambda_1\lambda_2
\end{aligned} \tag{3.9}$$

The P2 basis can be expressed in the following way:

$$\begin{aligned}
\psi &= \begin{pmatrix} 1 & 0 & 0 & 0 & -1 & -1 \\ 0 & 1 & 0 & -1 & 0 & -1 \\ 0 & 0 & 1 & -1 & -1 & 0 \\ 0 & 0 & 0 & 4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 0 & 0 & 4 \end{pmatrix} \begin{pmatrix} \lambda_1^2 \\ \lambda_2^2 \\ \lambda_3^2 \\ \lambda_2\lambda_3 \\ \lambda_1\lambda_3 \\ \lambda_1\lambda_2 \end{pmatrix} \\
&= \underbrace{\quad}_{A_{6 \times 6}} \underbrace{\quad}_R
\end{aligned} \tag{3.10}$$

With the help of the basis function, functions in each element can be approximated by:

$$u(x, y, t) = \sum_{j=1}^3 U_j \lambda_j(x, y) \tag{3.11}$$

$$\text{or } u(x, y, t) = \sum_{j=1}^6 U_j \psi_j(x, y) \tag{3.12}$$

where λ and ψ are the basis functions for P1 FEM and P2 FEM respectively, the number of degrees of freedom for P1 FEM is 3, and the number of degrees of freedom for P2 FEM is 6 in each element.

The following integral formula will be very useful for numerical implementations of the Finite Element method.

Proposition 3.3. *Let λ_1 , λ_2 , and λ_3 be the P1 finite element basis functions of element Ω_e with area Δ_e , and a , b , c be three nonnegative integers. Then*

$$\int_{\Omega_e} \lambda_1^a \lambda_2^b \lambda_3^c d\Omega_e = 2\Delta_e \frac{a!b!c!}{(a+b+c+2)!} \quad (3.13)$$

This is an exact integration and can be very useful beyond integrations of P1 basis functions, as P2 basis functions are constructed by P1 basis functions. The local mass matrix and stiffness matrix for P1 and P2 FEM can be calculated as following:

Local mass matrix for P1 FEM: $M_{i,j}^e = \int_{\Omega_e} \lambda_i \lambda_j d\Omega_e$

$$M_{i,j}^e = \frac{\Delta_e}{12} \begin{pmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{pmatrix} \quad (3.14)$$

Local mass matrix for P2 FEM: $M_{i,j}^e = \int_{\Omega_e} \psi_i \psi_j d\Omega_e$

According to (3.10), $M_{i,j}^e = \int_{\Omega_e} \psi_i \psi_j d\Omega_e = \int_{\Omega_e} \psi \psi^T d\Omega_e = A \int_{\Omega_e} R R^T d\Omega_e A^T$, thus

$$M_{i,j}^e = \frac{\Delta_e}{180} \begin{pmatrix} 6 & -1 & -1 & -4 & 0 & 0 \\ -1 & 6 & -1 & 0 & -4 & 0 \\ -1 & -1 & 6 & 0 & 0 & -4 \\ -4 & 0 & 0 & 32 & 16 & 16 \\ 0 & -4 & 0 & 16 & 32 & 16 \\ 0 & 0 & -4 & 16 & 16 & 32 \end{pmatrix} \quad (3.15)$$

Local stiffness matrix for P1 FEM: $K_{i,j}^e = \int_{\Omega_e} \nabla \lambda_i \cdot \nabla \lambda_j d\Omega_e$

$$K_{i,j}^e = \int_{\Omega_e} \nabla \lambda_i \cdot \nabla \lambda_j d\Omega_e = \frac{1}{4\Delta_e} (b_i b_j + c_i c_j) \quad (3.16)$$

Local stiffness matrix for P2 FEM: $K_{i,j}^e = \int_{\Omega_e} \nabla \psi_i \cdot \nabla \psi_j d\Omega_e$ are calculated in the following manner:

$$\frac{\partial \psi}{\partial x} = \frac{A}{2\Delta_e} \begin{pmatrix} 2b_1 & 0 & 0 \\ 0 & 2b_2 & 0 \\ 0 & 0 & 2b_3 \\ 0 & b_3 & b_2 \\ b_3 & 0 & b_1 \\ b_2 & b_1 & 0 \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{pmatrix} \quad (3.17)$$

$$\frac{\partial \psi}{\partial y} = \frac{A}{2\Delta_e} \begin{pmatrix} 2c_1 & 0 & 0 \\ 0 & 2c_2 & 0 \\ 0 & 0 & 2c_3 \\ 0 & c_3 & c_2 \\ c_3 & 0 & c_1 \\ c_2 & c_1 & 0 \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{pmatrix} \quad (3.18)$$

Denote

$$B = \begin{pmatrix} 2b_1 & 0 & 0 \\ 0 & 2b_2 & 0 \\ 0 & 0 & 2b_3 \\ 0 & b_3 & b_2 \\ b_3 & 0 & b_1 \\ b_2 & b_1 & 0 \end{pmatrix} \quad C = \begin{pmatrix} 2c_1 & 0 & 0 \\ 0 & 2c_2 & 0 \\ 0 & 0 & 2c_3 \\ 0 & c_3 & c_2 \\ c_3 & 0 & c_1 \\ c_2 & c_1 & 0 \end{pmatrix}$$

the stiffness matrix is actually:

$$\begin{aligned} K_{i,j}^e &= \int_{\Omega_e} \nabla \psi_i \cdot \nabla \psi_j d\Omega_e \\ &= \frac{1}{4\Delta_e^2} AB \int_{\Omega_e} \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{pmatrix} \begin{pmatrix} \lambda_1 & \lambda_2 & \lambda_3 \end{pmatrix} d\Omega_e B^T A^T \\ &\quad + \frac{1}{4\Delta_e^2} AC \int_{\Omega_e} \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \end{pmatrix} \begin{pmatrix} \lambda_1 & \lambda_2 & \lambda_3 \end{pmatrix} d\Omega_e C^T A^T \end{aligned}$$

while the integral has been calculated as the mass matrix for P1 FEM. Finally the local stiffness matrix for P2 FEM is:

$$K_{i,j}^e = \frac{1}{48\Delta_e} \left(AB \begin{pmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{pmatrix} B^T A^T + AC \begin{pmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{pmatrix} C^T A^T \right) \quad (3.19)$$

3.1.2 Matrix assembly

Now we want to obtain the global matrix by assembling all the local matrices. We consider assembling the P2 stiffness matrix for illustration and a similar procedure can be applied to mass matrix as well. By definition, suppose N is the degree of freedom for P2 FEM. For $0 \leq i, j \leq N - 1$, the global stiffness matrix $K = (K_{i,j})_{N \times N}$ is:

$$K_{i,j} = \int_{\Omega} \nabla \psi_i \cdot \nabla \psi_j d\Omega = \sum_{\Omega_e \in \mathcal{T}_h} \int_{\Omega_e} \nabla \psi_i \cdot \nabla \psi_j d\Omega_e \quad (3.20)$$

The assemble process is to "glue" the quantity associated with the local index to the global numbering. We have already calculated the local stiffness matrix as shown in (3.19), so a standard subroutine to compute the global stiffness matrix is:

Algorithm 1 Standard way of assembly stiffness matrix

```
1: for  $t = 0$  to  $Th.nt - 1$  do
2:   calculate local stiffness matrix  $K^e$ 
3:   for  $i = 0$  to 5 do
4:     for  $j = 0$  to 5 do
5:        $K(Vh2(t, i), Vh2(t, j)) = K(Vh2(t, i), Vh2(t, j)) + K^e(i, j)$ 
6:     end for
7:   end for
8: end for
```

where $Th.nt$ is the number of elements and $Vh2(t, i)$ records the global numbering of the i th degree of freedom in element t . Notice that the global stiffness matrix K needs $O(N^2)$ storage, and it can be seen in the following chapters that by using operator splitting scheme, we need to update quite a few matrices due to discontinuous coefficients. Thus the standard way of assembly may not be very efficient, because the computer will be out of memory quickly when N is big. The good news is that when we perform FEM discretization, the matrices obtained are sparse matrices, see [53], [10]. So even the global matrix is $N \times N$, we

only need to storage the nonzero entries. We can solve this storage problem by storing the matrix K into a sparse matrix. However, line 5 in Algorithm 1 is quite expensive, because accessing an element of a sparse matrix requires searching for the index vectors to find the corresponding nonzero entry. We can solve this problem by recording the index and nonzero entries using three vectors $I(36 \times Th.nt)$, $J(36 \times Th.nt)$, and $S(36 \times Th.nt)$ in the FOR loop in line 5 of Algorithm 1 and building the sparse matrix outside the loop, where I and J record the row and column number of the nonzero entry and S record the nonzero entry, i.e.

Algorithm 2 Assembly stiffness matrix

```

1:  $index = 0$ 
2: for  $t = 0$  to  $Th.nt - 1$  do
3:   calculate local stiffness matrix  $K^e$ 
4:   for  $i = 0$  to 5 do
5:     for  $j = 0$  to 5 do
6:        $I(index) = Vh2(t, i)$ 
7:        $J(index) = Vh2(t, j)$ 
8:        $S(index) = K^e(i, j)$ 
9:        $index = index + 1$ 
10:    end for
11:  end for
12: end for
13: matrix  $K$ 
14:  $K = [I, J, S]$ 

```

Although this algorithm still needs to loop over all the elements, it is good enough for us because the computational complexity has been improved from $O(N^2)$ to $O(N \log N)$. Matlab users can still improve the efficiency by vectorization of assembling, during which the outer large FOR loop in line 2 of Algorithm 2 can be avoided.

3.2 Numerical solution of the Navier Stokes equations

3.2.1 Choice of interpolation functions

To implement Finite Element discretization, we need to choose proper interpolation function spaces for velocity and pressure fields, and equal order Finite Element interpolation. For example, the P1 finite interpolation for both of velocity and pressure fields is the most tempting one due to its simplicity. However, this is unfortunately not always true, even when we are using projection method. This problem was first introduced when studying the stability of mixed method of steady Stokes equation, which leads to the theory called Ladyzhenskaya-Babuska-Brezzi(LBB) compatibility condition [23] [1], [4]. LBB condition states that the interpolation spaces of velocity and pressure fields can not be chosen arbitrarily. They must satisfy some inf-sup conditions to get a stable finite element approximation. Generally, there are three ways to overcome this numerical difficulties. The first option is the penalty method [22], [70], [56]. By introducing a parameter associated with pressure field, it completely replaces the incompressibility condition such that the pressure field is not a primitive variable in the momentum equation. The penalty method is attractive because it eliminates the incompressibility from the variational formulation, and thus it reduces the number of variables. However, like most parametrization techniques, the success of the penalty method mainly depends on the choice of the parameters. Another way to overcome the LBB condition are the stabilization techniques. Inspired from stabilized formulation, stabilization techniques can be implemented by a modification of the weak form of the incompressibility condition such that the resulting algebraic system is a positive definiteness one, which gives sound numerical results. Some of the stabilization techniques, for instance Galerkin/Least-squares(GLS) [14], only require equal order interpolations for velocity and pressure fields for the Galerkin formulation to become stable. Among all of

the three options, finding possible pairs of velocity and pressure fields in stable elements seems the most tempting way to conquer the LBB condition, because well-chosen velocity and pressure pairs in stable finite element can eliminate this condition completely. As we want to use continuous pressure finite element approximation, the elements proposed by Taylor and Hood [17], and Bercovier-Pironneau [41], are two natural ones. We show in Figure (3.2) and (3.3) Taylor-Hood and Bercovier-Pironneau triangle elements respectively.

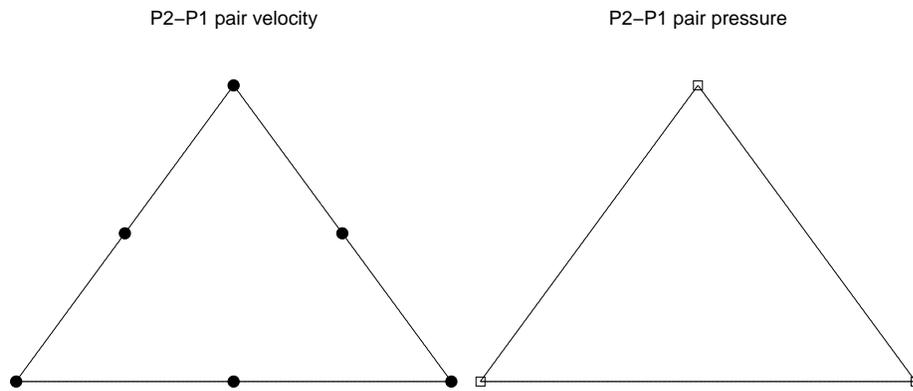


Figure 3.2: Taylor-Hood Element

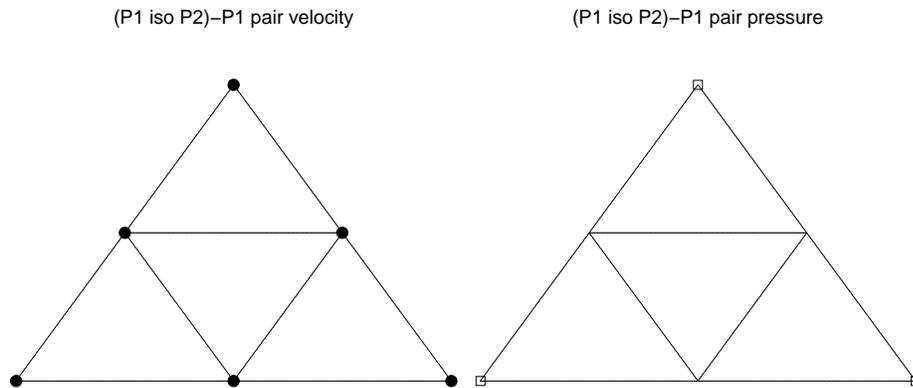


Figure 3.3: Bercovier-Pironneau Element

In Taylor-Hood elements, the velocity field is approximated by piecewise quadratic functions and pressure is approximated by piecewise linear functions. In these elements, only the base elements are used and in each element, each velocity component is associated with 6 degrees of freedom and the pressure field has 3 degrees of freedom. Taylor-Hood elements have been proved to be stable, and they converge at the optimal rate [46]. Bercovier-Pironneau elements follow the similar ideas as Taylor-Hood. The pressure space is still the same, but each component of velocity is approximated by a linear function in each of the four subtriangles. Bercovier and Pironneau pointed out this is a stable element with linear convergency. Both elements can be applied when the triangulation is made of parallelograms. There are many other stable elements that satisfy LBB conditions, for example, the Crouzeix-Raviart elements [45] and mini elements proposed by Arnold, Brezzi and Fortin [12]. Guermond and Quartapelle further studied of velocity and pressure pairs associated with fractional step project methods. They concluded that in order to obtain non-oscillatory results, the velocity and pressure pair must satisfy the LBB condition [43], [44]. We present the Taylor-Hood elements with P2-P1 pair of velocity and pressure fields for the rest of this dissertation for illustration.

3.2.2 Time discretization: fractional step projection scheme

Before we proceed to solve the Navier Stokes equations, we briefly introduce fractional step schemes. Fractional step schemes have been thoroughly studied by Marchuk [28] [29] and Yanenko [66]. Detailed analysis, including the overall accuracy of the scheme, can be found in specialized textbooks by Marchuk [28] [29], Yanenko [66], Quarteroni and Valli [61], Glowinski and Le Tallec, et al. [26]. In general, fractional step schemes can be performed either at differential or algebraic level. No matter in which level, however, it follows

the so-called divide and conquer strategy, meaning decompose system of PDEs into sub-problems and treat them individually using specialized numerical algorithms, namely:

Definition 3.4. For a initial boundary value problem

$$\begin{aligned}\frac{\partial u}{\partial t} + \mathcal{L}u &= f \quad \text{in } \Omega \times [0, T] \\ u|_{t=0} &= u^{(0)} \quad \text{in } \Omega\end{aligned}$$

fractional step scheme means

$$\mathcal{L} = \sum_{s=1}^S \mathcal{L}_s$$

where \mathcal{L}_s can represent different physical phenomena, for example, convection, diffusion reaction etc.

However, one should notice that in case of splitting, the sequential treatment of different physical processes requires a splitting of boundary conditions as well, so that each individual problem has reasonable boundary data. For example, we can only impose boundary data on the inflow part of the boundary for convection problem. Besides, the fractional step scheme is also sensitive with the source term. Details of those issues will be presented again in the following section.

In our particular case, two numerical difficulties can not be avoidable in application of the standard Galerkin Finite Element method. The first one is due to the presence of fluid nonlinear convection term in the Navier Stokes equations. It has been known that the standard Galerkin formulation is unstable when the problem is convection dominated with high Reynolds numbers. We need more stabilization techniques when high Reynolds number presents. In fact, we can linearize the nonlinear convection term first and convect

the problem using fixed known transport velocity from previous time step. Alternatively, we can even treat the convective term completely explicit on the right-hand side. But these techniques require smaller time step sizes when the nonlinearity effect is strong. Notice we need higher order time schemes in coupling time integration and spatial representation, since even an accurate spatial representation can be useless when the time integration can not propagate the information along the characteristics. In this case, we use a fourth order Taylor-Galerkin scheme, named TTG-4A method. It is attractive because it is an explicit scheme, and the scheme itself can treat the nonlinearity without linearization techniques. Another very important issue in approximation of the Navier Stokes equations are related to the incompressibility condition of velocity field. Since pressure acts like a Lagrangian multiplier in the Navier Stokes equations, this leads to a saddle point problem of the resulting variational formulation. In order to solve the resulting algebraic system, a proper choice of interpolation functions of velocity and pressure field is needed. Besides, velocity pressure coupling in the Navier Stokes equations should be underlined. One can solve velocity and pressure simultaneously, but this will come up with large algebraic systems. It will reduce computational cost significantly if the velocity and pressure approximations are decoupled. This leads to projection methods. The principle of a projection method is to compute the velocity and pressure fields separately through the computation of an intermediate velocity field, and then project the the velocity field onto the subspace of the solenoidal vector functions. The projection method is based on the following Ladyzhenskaya orthogonal decomposition theorem, which is derivation of Helmholtz decomposition principle.

Theorem 3.5. Any vector field \vec{w} defined in Ω admits the unique orthogonal decomposition

$$\vec{w} = \vec{v} + \nabla\phi \tag{3.21}$$

where \vec{v} is a solenoidal field with zero normal component on the domain boundary (i.e. $\nabla \cdot \vec{v} = 0$ in Ω , and $\vec{n} \cdot \vec{v} = 0$ on Σ where Σ is the boundary of Ω), and ϕ is a scalar function.

Proof. We begin the proof by checking the existence of the scalar function ϕ . Since $\vec{w} = \vec{v} + \nabla\phi$, we have the following Neumann problem

$$\begin{cases} \Delta\phi = \nabla \cdot \vec{w} & \text{in } \Omega \\ \vec{n} \cdot \nabla\phi = \vec{n} \cdot \nabla\vec{w} & \text{on } \Sigma \end{cases}$$

The solvability of this Neumann problem is satisfied due to divergence theorem, namely, $\int_{\Omega} \nabla \cdot \vec{w} d\Omega = \int_{\Sigma} \vec{n} \cdot \vec{w} d\Sigma$, so the solution ϕ to this problem exists and is defined up to an arbitrary constant.

Next we check the orthogonality of the decomposition. Recall the identity $\nabla \cdot (\vec{v}\phi) = (\nabla \cdot \vec{v})\phi + \vec{v} \cdot \nabla\phi$, and the divergence theorem, we have

$$\int_{\Omega} \vec{v} \cdot \nabla\phi d\Omega = \int_{\Omega} \nabla \cdot (\vec{v}\phi) d\Omega = \int_{\Sigma} \phi \vec{v} \cdot \vec{n} d\Sigma = 0$$

Finally, we use the orthogonality to prove the uniqueness. Suppose $\vec{w} = \vec{v}_1 + \nabla\phi_1 = \vec{v}_2 + \nabla\phi_2$, then

$$\vec{v}_1 - \vec{v}_2 + \nabla(\phi_1 - \phi_2) = 0$$

Take the dot product with $\vec{v}_1 - \vec{v}_2$ and integrate on Ω to get

$$0 = \int_{\Omega} |\vec{v}_1 - \vec{v}_2|^2 + (\vec{v}_1 - \vec{v}_2) \cdot \nabla(\phi_1 - \phi_2) d\Omega = \int_{\Omega} |\vec{v}_1 - \vec{v}_2|^2 d\Omega$$

where the second equality is due to the orthogonality. It follows that $\vec{v}_1 = \vec{v}_2$, $\nabla\phi_1 = \nabla\phi_2$, and that means $\phi_1 = \phi_2 + \text{constant}$. This completes the proof. \square

Notice our problem is more complicated due to the presence of an interface. We can still use the fractional step projection scheme because we capture the interfacial condition as a boundary condition. However, this involves more careful investigation when we do splitting. We illustrate three typical forms of fractional step projection methods known as the Chorin Teman projection method, the three step fractional step projection method and the Van Kan's projection scheme. Each of the projection schemes have linearized and nonlinearized versions due to discontinuous density and viscosity.

3.2.3 Numerical solutions for the subproblems

As mentioned before, splitting the incompressibility condition is tempting. This naturally leads to the projection scheme, during which we can decompose the complex problem into at least two simpler ones, i.e. the momentum equation and the pressure Poisson equation. Regarding the convective term, the fractional step projection method can be further divided into the two-step method and three-step method. In this dissertation, we investigate both of them, as well as another two-step method named Van Kan's projection scheme, in detail. The good news is as long as supplements of boundary conditions can be specified for each subproblem, we have specialized numerical algorithms to solve those

simpler subproblems efficiently. We begin with the analysis of the two-step projection scheme. The three-step projection scheme and Van Kan's projection scheme follow similar ideas and will be included as well.

3.2.3.1 Chorin Teman projection scheme

The Chorin Teman projection scheme consists of solving the convective term and viscous term together first in the Navier Stokes momentum equation, followed by considering gradient of the pressure term and the incompressibility condition.

Consider the following Navier Stokes equations in domain $\Omega \times [0, T]$:

$$\rho(\vec{x})\left(\frac{\partial \vec{u}}{\partial t} + (\vec{u} \cdot \nabla)\vec{u}\right) - \nabla \cdot (\mu(\vec{x})(\nabla \vec{u} + \nabla \vec{u}^T)) + \nabla p = \rho(\vec{x})\vec{g} \quad \text{in } \Omega \times [0, T] \quad (3.22)$$

$$\nabla \cdot \vec{u} = 0 \quad (3.23)$$

$$[\vec{u}]_{\Gamma} = \vec{0}, \quad -[-pI + \mu(\vec{x})(\nabla \vec{u} + \nabla \vec{u}^T)]|_{\Gamma} \cdot \vec{n} = \sigma \kappa \vec{n} \quad (3.24)$$

$$\vec{u}(\vec{x}, 0) = \vec{u}^{(0)}(\vec{x}) \quad \text{in } \Omega \quad (3.25)$$

$$\vec{u}(\vec{x}, t) = \vec{b} \quad \text{on } \Sigma \quad (3.26)$$

where all symbols have been defined in previous chapters with Γ is the interface, Σ is the boundary of the domain, equation (3.22) is the conservation of momentum, equation (3.23) is the incompressibility condition, equations (3.24) are interfacial conditions, equation (3.25) is the initial data, and equation (3.26) is the boundary condition, $\rho(\vec{x}) = \rho_1$, in Ω_1 , $\rho(\vec{x}) = \rho_2$, in Ω_2 , $\mu(\vec{x}) = \mu_1$, in Ω_1 , $\mu(\vec{x}) = \mu_2$, in Ω_2 .

The compatibility condition has to be satisfied due to divergence theorem as

$$0 = \int_{\Omega} \nabla \cdot \vec{u} d\Omega = \int_{\Sigma} \vec{u} \cdot \vec{n} d\Sigma = \int_{\Sigma} \vec{b} \cdot \vec{n} d\Sigma \quad (3.27)$$

Chorin Teman begins with solving an intermediate velocity field without the incompressibility condition and the pressure term. The first task is to solve the following system:

$$\rho(\vec{x})\left(\frac{\partial \vec{u}}{\partial t} + (\vec{u} \cdot \nabla)\vec{u}\right) - \nabla \cdot (\mu(\vec{x})(\nabla \vec{u} + \nabla \vec{u}^T)) = \rho(\vec{x})\vec{g} \quad \text{in } \Omega \times [t_n, t_{n+1}] \quad (3.28)$$

$$[\vec{u}]_\Gamma = \vec{0}, \quad -[\mu(\vec{x})(\nabla \vec{u} + \nabla \vec{u}^T)]|_\Gamma \cdot \vec{n} = \sigma \kappa \vec{n} \quad (3.29)$$

$$\vec{u}(\vec{x}, t_n) = \vec{u}^{(n)}(\vec{x}) \quad \text{in } \Omega \quad (3.30)$$

$$\vec{u} = \vec{b} \quad \text{on } \Sigma \quad (3.31)$$

$$\implies \vec{u}^{n+\frac{1}{2}}$$

Notice we omit the pressure term in the momentum equation and in the interfacial condition, during which the jump of viscous stress is balanced with the surface tension force. We only prescribe the Dirichlet boundary condition because we are solving a viscous problem in this step. This convection diffusion problem remains to be solved with proper treatment of the discontinuous density and viscosity fields, the nonlinear convective term, and the surface tension effect. Following Smolianski's work [52], this convection diffusion equation can be further split into two separated subproblems: a convection problem and a viscous diffusion problem; the surface tension effect is only associated with the viscous diffusion process. Using proper time discretization technique this way, an intermediate velocity field $\vec{u}^{n+\frac{1}{2}}$ can be found. Next, use the intermediate velocity field as an initial condition, and decompose it into the sum of a solenoidal velocity field \vec{u}^{n+1} and the gradient of unknown pressure, which leads to the second step of the Chorin Teman projection method:

$$\frac{\partial \vec{u}}{\partial t} + \frac{1}{\rho(\vec{x})} \nabla p = \vec{0} \quad \text{in } \Omega \times [t_n, t_{n+1}] \quad (3.32)$$

$$\nabla \cdot \vec{u} = 0 \quad (3.33)$$

$$[\vec{u}]_{\Gamma} \cdot \vec{n} = \vec{0}, \quad [p]_{\Gamma} = 0 \quad (3.34)$$

$$\vec{u}(\vec{x}, t_n) = \vec{u}^{n+\frac{1}{2}} \quad \text{in } \Omega \quad (3.35)$$

$$\vec{u} \cdot \vec{n} = \vec{b} \cdot \vec{n} \quad \text{on } \Sigma \quad (3.36)$$

$$\implies \vec{u}^{n+1}, p^{n+1}$$

We loop over time to get the approximation of velocity and pressure fields. It should be pointed out that we can only prescribe boundary condition on the normal component of velocity, and this is a consequence of the Ladyzhenskaya orthogonal decomposition theorem. This verifies the physics as we are dealing with an inviscid flow problem in the second step.

Considering for instance the following time discretiation scheme. The second step can be formulated as a Poisson type equation for pressure p^{n+1} .

$$\frac{\vec{u}^{n+1} - \vec{u}^{n+\frac{1}{2}}}{\Delta t} = -\frac{1}{\rho(\vec{x})} \nabla p^{n+1} \quad (3.37)$$

$$\nabla \cdot \vec{u}^{n+1} = 0 \quad (3.38)$$

$$\vec{u}^{n+1} \cdot \vec{n} = \vec{b} \cdot \vec{n} \quad (3.39)$$

where $\Delta t = t_{n+1} - t_n$ is the time step and $n = 0, 1, 2, \dots$. Applying divergence operator on both sides of equation (3.37), we have

$$\frac{\nabla \cdot \vec{u}^{n+1} - \nabla \cdot \vec{u}^{n+\frac{1}{2}}}{\Delta t} = -\nabla \cdot \left(\frac{1}{\rho(\vec{x})} \nabla p^{n+1} \right)$$

which is

$$\nabla \cdot \left(\frac{1}{\rho(\vec{x})} \nabla p^{n+1} \right) = \frac{1}{\Delta t} \nabla \cdot \vec{u}^{n+\frac{1}{2}}$$

The boundary condition is obtained by taking the normal component of equation (3.37), and taking account of boundary condition $\vec{u}^{n+\frac{1}{2}}|_{\Sigma} = \vec{b}$ on the first step and $\vec{u}^{n+1} \cdot \vec{n}|_{\Sigma} = \vec{b} \cdot \vec{n}$ on the second step. Thus, the pressure p^{n+1} is the solution of the following Poisson-type Neumann problem:

$$\begin{aligned} \nabla \cdot \left(\frac{1}{\rho(\vec{x})} \nabla p^{n+1} \right) &= \frac{1}{\Delta t} \nabla \cdot \vec{u}^{n+\frac{1}{2}} \\ \nabla p^{n+1} \cdot \vec{n}|_{\Sigma} &= 0 \end{aligned} \tag{3.40}$$

The solvability condition for this Neumann problem is

$$0 = \int_{\Sigma} \frac{1}{\rho(\vec{x})} \nabla p^{n+1} \cdot \vec{n} d\Sigma = \int_{\Omega} \nabla \cdot \left(\frac{1}{\rho(\vec{x})} \nabla p^{n+1} \right) d\Omega = \int_{\Omega} \nabla \cdot \vec{u}^{n+\frac{1}{2}} d\Omega$$

and together with the compatibility condition (3.27), this can be justified as another exercise of divergence theorem

$$\int_{\Omega} \nabla \cdot \vec{u}^{n+\frac{1}{2}} d\Omega = \int_{\Sigma} \vec{u}^{n+\frac{1}{2}} \cdot \vec{n} d\Sigma = \int_{\Sigma} \vec{b} \cdot \vec{n} d\Sigma = 0$$

Since the solution of the Neumann problem (3.40) is defined up to an additive constant, a reference value of pressure must be specified at an arbitrary pressure node. Besides,

the boundary condition on (3.40) is obtained after time discretization of the original system (3.32)-(3.36), so the exact pressure does not necessary satisfy the Neumann boundary condition; this has been studied by Gresho and Sani [48]. Gresho [15] and Temam [55] proposed this non-physical homogeneous Neumann boundary condition may result in a spurious pressure boundary layer of width $O(\sqrt{\nu\Delta t})$, where $\nu = \mu/\rho$ is the kinematic viscosity. This is one drawback of the projection scheme. Another thing that should catch our attention is the Laplace operator. We might need to assume higher regularity of the solution due to this second order operator. In the context of using a Finite Element method to solve this Neumann problem, the presence of a discontinuous density field need more investigation compared with single flow problem, where we need to do nothing but constructing the lumped mass matrix. Once the pressure p^{n+1} is obtained, the velocity field can be given via equation (3.37), namely,

$$\vec{u}^{n+1} = \vec{u}^{n+\frac{1}{2}} - \frac{1}{\rho(\vec{x})} \Delta t \nabla p^{n+1} \quad (3.41)$$

We proceed with the Finite Element framework for the two subproblems.

Step 1. Navier Stokes convection diffusion step

The associated spatial discretization for subproblem (3.28) to (3.31) is given as follows:

Find $\vec{u} \in H^1(\Omega) \times H^1(\Omega)$ and $\vec{u} = \vec{b}$ on Σ , such that,

$$\int_{\Omega} \rho(\phi) \frac{\partial \vec{u}}{\partial t} \cdot \vec{w} d\Omega + \int_{\Omega} \rho(\phi) \vec{u} \cdot \nabla \vec{u} \cdot \vec{w} d\Omega + \int_{\Omega} \mu(\phi) (\nabla \vec{u} + \nabla \vec{u}^T) : \nabla \vec{w}^T d\Omega = \int_{\Omega} \rho(\phi) \vec{g} \cdot \vec{w} d\Omega + \int_{\Gamma} \kappa \sigma \vec{n} \cdot \vec{w} \quad (3.42)$$

for $\forall \vec{w} \in H^1(\Omega) \times H^1(\Omega)$ such that $\vec{w} = \vec{0}$ on Σ .

Remark 3.6. *The interfacial condition can be expressed as a line integral in Finite Element framework.*

Remark 3.7. *Spatial variables \vec{x} in discontinuous coefficients $\rho(\vec{x})$ and $\mu(\vec{x})$ are replaced by $\rho(\phi)$ and $\mu(\phi)$ respectively simply because the evaluation of density and viscosity depends on the interface position Γ . Recall in the Level Set method, we use a signed distance function ϕ to represent the interface:*

$$\Gamma = \{\vec{x} | \phi(\vec{x}, t) = 0\} \quad \text{for } \forall t > 0$$

ϕ is a time-dependent scalar function. We underline the importance of this function. Evaluating it at correct time level is crucial for the whole problem, see an important remark in the work of Groß [16].

The spatial discretization above is actually:

$$M(t)U'(t) + (C(U(t), t) + K(t))U(t) = b(t) \tag{3.43}$$

where M, C, K are the mass, convection and diffusion part respectively, and the right-hand side b is the body force. We rule out the surface tension force for the moment for simplicity, but we will include this term later.

Now we want to solve an ODE system of the following type:

$$U'(t) = M^{-1}(t)T(U(t), t) \tag{3.44}$$

where $T(U(t), t) = b(t) - (C(U(t), t) + K(t))U(t)$.

Based on Groß's work [16], we proposed the following linearized and non-linearized schemes:

Linearized method

Linearized method with Crank-Nicolson method ($\theta = 0.5$) is given as follows:

$$\frac{U^{n+\frac{1}{2}} - U^n}{\Delta t} = \theta M_n^{-1} T(U^{n+\frac{1}{2}}, t^{n+\frac{1}{2}}) + (1 - \theta) M_n^{-1} T(U^n, t^n) \quad (3.45)$$

Multiply M_n on both sides and recall the notation of T . This leads to the following scheme:

$$(M_n + \theta \Delta t (C_n + K_n)) U^{n+\frac{1}{2}} = (M_n - (1 - \theta) \Delta t (C_n + K_n)) U^n + \theta \Delta t b^{n+\frac{1}{2}} + (1 - \theta) \Delta t b^n \quad (3.46)$$

Subscripts for the matrix M, C and K denote the dependence of Level Set function at different time levels. For example, M_n means that using ϕ^n , namely the Level Set function from previous time step, to evaluate the mass matrix.

Non-linearized method

Non-linearized method means to evaluate the right-hand side of ODE in different time levels, which is:

$$\frac{U^{n+\frac{1}{2}} - U^n}{\Delta t} = \theta M_{n+\frac{1}{2}}^{-1} T(U^{n+\frac{1}{2}}, t^{n+\frac{1}{2}}) + (1 - \theta) M_n^{-1} T(U^n, t^n) \quad (3.47)$$

Multiply $M_{n+\frac{1}{2}}$ on both sides and recall the notation of T , it gives:

$$(M_{n+\frac{1}{2}} + \theta \Delta t (C_{n+\frac{1}{2}} + K_{n+\frac{1}{2}})) U^{n+\frac{1}{2}} = M_{n+\frac{1}{2}} (U^n + (1 - \theta) \Delta t M_n^{-1} (b^n - (C_n + K_n) U^n)) + \theta \Delta t b^{n+\frac{1}{2}}$$

In other words, we need to solve the following system:

$$(M_{n+\frac{1}{2}} + \theta\Delta t(C_{n+\frac{1}{2}} + K_{n+\frac{1}{2}}))U^{n+\frac{1}{2}} = M_{n+\frac{1}{2}}(U^n + (1-\theta)\Delta tV^n) + \theta\Delta tb^{n+\frac{1}{2}} \quad (3.48)$$

where V^n is the solution of the following equation:

$$M_nV^n = b^n - (C_n + K_n)U^n \quad (3.49)$$

Remark 3.8. *It should be noticed that if the convection term dominates, some stabilizations are needed for that term. For example, the counter part of $\theta\Delta tC_nU^{n+\frac{1}{2}} + (1-\theta)\Delta tC_nU^n$ in the linearized version, namely the following term*

$$\theta\Delta t \int_{\Omega} \rho(\phi^n)\bar{u}^n \cdot \nabla\bar{u}^{n+\frac{1}{2}} \cdot \bar{w}d\Omega + (1-\theta)\Delta t \int_{\Omega} \rho(\phi^n)\bar{u}^n \cdot \nabla\bar{u}^n \cdot \bar{w}d\Omega \quad (3.50)$$

must be stabilized. The following second-order approximation could be an option for stabilization.

$$\begin{aligned} & \Delta t \int_{\Omega} \rho(\phi^n)\bar{u}^n \cdot \nabla\bar{u}^n \cdot \bar{w}d\Omega \\ & + (1-\theta)\frac{\Delta t^2}{2} \int_{\Omega} \rho(\phi^n)(\bar{u}^n \cdot \nabla\bar{u}^n)(\bar{u}^n \cdot \nabla\bar{w})d\Omega \\ & + \theta\frac{\Delta t^2}{2} \int_{\Omega} \rho(\phi^n)(\bar{u}^n \cdot \nabla\bar{u}^{n+\frac{1}{2}})(\bar{u}^n \cdot \nabla\bar{w})d\Omega \end{aligned} \quad (3.51)$$

Remark 3.9. *If the viscosity μ is small, standard grad-div stabilization technique can be*

used to calibrate the velocity profile in the momentum equation.

The product $(\nabla \vec{u} + \vec{u}^T) : \nabla \vec{w}^T$ in (3.42) is,

$$(\nabla \vec{u} + \vec{u}^T) : \nabla \vec{w}^T = \text{tr} \left[\begin{pmatrix} 2 \frac{\partial u_1}{\partial x} & \frac{\partial u_1}{\partial y} + \frac{\partial u_2}{\partial x} \\ \frac{\partial u_1}{\partial y} + \frac{\partial u_2}{\partial x} & 2 \frac{\partial u_2}{\partial y} \end{pmatrix} \begin{pmatrix} \frac{\partial w_1}{\partial x} & \frac{\partial w_2}{\partial x} \\ \frac{\partial w_1}{\partial y} & \frac{\partial w_2}{\partial y} \end{pmatrix} \right]$$

It might be a good idea to see the weak formulation explicitly for each component of velocity field. To do so, we need to take \vec{w} as $\vec{w} = \begin{pmatrix} w_1 \\ 0 \end{pmatrix}$, and $\vec{w} = \begin{pmatrix} 0 \\ w_2 \end{pmatrix}$ respectively, and add back the surface tension force. The weak formulation for each component of velocity field in the linearized version reads:

Find $u_1^{n+\frac{1}{2}} \in H^1(\Omega)$ and $u_1^{n+\frac{1}{2}} = b_1$ on Σ , such that,

$$\begin{aligned} & \int_{\Omega} \rho(\phi^n) u_1^{n+\frac{1}{2}} w_1 d\Omega + \theta \Delta t \int_{\Omega} \rho(\phi^n) \left(u_1^n \frac{\partial u_1^{n+\frac{1}{2}}}{\partial x} + u_2^n \frac{\partial u_1^{n+\frac{1}{2}}}{\partial y} \right) w_1 d\Omega \\ & + \theta \Delta t \int_{\Omega} \mu(\phi^n) \left(\frac{\partial u_1^{n+\frac{1}{2}}}{\partial x} \frac{\partial w_1}{\partial x} + \frac{\partial u_1^{n+\frac{1}{2}}}{\partial y} \frac{\partial w_1}{\partial y} + \frac{\partial u_1^{n+\frac{1}{2}}}{\partial x} \frac{\partial w_1}{\partial x} + \frac{\partial u_2^{n+\frac{1}{2}}}{\partial x} \frac{\partial w_1}{\partial y} \right) d\Omega \\ & = \int_{\Omega} \rho(\phi^n) u_1^n \cdot w_1 d\Omega \\ & - (1 - \theta) \Delta t \int_{\Omega} \rho(\phi^n) \left(u_1^n \frac{\partial u_1^n}{\partial x} + u_2^n \frac{\partial u_1^n}{\partial y} \right) w_1 d\Omega \\ & - (1 - \theta) \Delta t \int_{\Omega} \mu(\phi^n) \left(\frac{\partial u_1^n}{\partial x} \frac{\partial w_1}{\partial x} + \frac{\partial u_1^n}{\partial y} \frac{\partial w_1}{\partial y} + \frac{\partial u_1^n}{\partial x} \frac{\partial w_1}{\partial x} + \frac{\partial u_2^n}{\partial x} \frac{\partial w_1}{\partial y} \right) d\Omega \\ & + \theta \Delta t \int_{\Omega} \rho(\phi^{n+1}) g_1 w_1 d\Omega + (1 - \theta) \Delta t \int_{\Omega} \rho(\phi^n) g_1 w_1 d\Omega \\ & + \Delta t \int_{\Gamma} f_1 w_1 d\Gamma \end{aligned} \tag{3.52}$$

for $\forall w_1 \in H^1(\Omega)$ such that $w_1 = 0$ on Σ .

Find $u_2^{n+\frac{1}{2}} \in H^1(\Omega)$ and $u_2^{n+\frac{1}{2}} = b_2$ on Σ , such that,

$$\begin{aligned}
& \int_{\Omega} \rho(\phi^n) u_2^{n+\frac{1}{2}} w_2 d\Omega + \theta \Delta t \int_{\Omega} \rho(\phi^n) \left(u_1^n \frac{\partial u_2^{n+\frac{1}{2}}}{\partial x} + u_2^n \frac{\partial u_2^{n+\frac{1}{2}}}{\partial y} \right) w_2 d\Omega \\
& + \theta \Delta t \int_{\Omega} \mu(\phi^n) \left(\frac{\partial u_2^{n+\frac{1}{2}}}{\partial x} \frac{\partial w_2}{\partial x} + \frac{\partial u_2^{n+\frac{1}{2}}}{\partial y} \frac{\partial w_2}{\partial y} + \frac{\partial u_1^{n+\frac{1}{2}}}{\partial y} \frac{\partial w_2}{\partial x} + \frac{\partial u_2^{n+\frac{1}{2}}}{\partial y} \frac{\partial w_2}{\partial y} \right) d\Omega \\
& = \int_{\Omega} \rho(\phi^n) u_2^n \cdot w_2 d\Omega \\
& - (1 - \theta) \Delta t \int_{\Omega} \rho(\phi^n) \left(u_1^n \frac{\partial u_2^n}{\partial x} + u_2^n \frac{\partial u_2^n}{\partial y} \right) w_2 d\Omega \\
& - (1 - \theta) \Delta t \int_{\Omega} \mu(\phi^n) \left(\frac{\partial u_2^n}{\partial x} \frac{\partial w_2}{\partial x} + \frac{\partial u_2^n}{\partial y} \frac{\partial w_2}{\partial y} + \frac{\partial u_1^n}{\partial y} \frac{\partial w_2}{\partial x} + \frac{\partial u_2^n}{\partial y} \frac{\partial w_2}{\partial y} \right) d\Omega \\
& + \theta \Delta t \int_{\Omega} \rho(\phi^{n+1}) g_2 w_2 d\Omega + (1 - \theta) \Delta t \int_{\Omega} \rho(\phi^n) g_2 w_2 d\Omega \\
& + \Delta t \int_{\Gamma} f_2 w_2 d\Gamma \tag{3.53}
\end{aligned}$$

for $\forall w_2 \in H^1(\Omega)$ such that $w_2 = 0$ on Σ .

where $\vec{f} = (f_1, f_2)^T = \kappa \sigma \vec{n}$. (3.52) and (3.53) use the Level Set function from previous time step to evaluate the discontinuous coefficients, and this is very important for the linearized version to obtain correct solutions.

Remark 3.10. *We use the Crank-Nicolson scheme for time integration here. In general, for the one-step θ family scheme, $\theta \in [0, 1]$. $\theta = 0$ is the explicit Euler scheme. Notice for the explicit scheme, u_1 and u_2 are fully decoupled. They can be solved independently since the second and third terms on (3.52), and (3.53) disappear completely. The explicit scheme requires very little time step in order to get stable results, especially when the viscous term dominates the problem, for example creeping flows. Thus the implicit schemes, namely $\theta \in (0, 1]$ are preferred in this step. For $\theta = \frac{1}{2}$, it is the Crank-Nicolson scheme, and for $\theta = 1$, it is the implicit Euler scheme. For $\theta \geq \frac{1}{2}$, the scheme is unconditionally stable.*

But the second and third terms on (3.52), and (3.53) need more attention, because the velocity field can not be solved separately due to the presence of $\nabla \bar{u}^T$ term. One can still make the velocity field decoupled by simply approximating the $\nabla \bar{u}^T$ term fully explicitly, during which $\frac{\partial u_1^{n+\frac{1}{2}}}{\partial x} \frac{\partial w_1}{\partial x} + \frac{\partial u_2^{n+\frac{1}{2}}}{\partial x} \frac{\partial w_1}{\partial y}$ and $\frac{\partial u_1^{n+\frac{1}{2}}}{\partial y} \frac{\partial w_2}{\partial x} + \frac{\partial u_2^{n+\frac{1}{2}}}{\partial y} \frac{\partial w_2}{\partial y}$ term are approximated by $\frac{\partial u_1^n}{\partial x} \frac{\partial w_1}{\partial x} + \frac{\partial u_2^n}{\partial x} \frac{\partial w_1}{\partial y}$ and $\frac{\partial u_1^n}{\partial y} \frac{\partial w_2}{\partial x} + \frac{\partial u_2^n}{\partial y} \frac{\partial w_2}{\partial y}$ respectively. Then they moved to the right-hand side of (3.52), and (3.53). Like the explicit schemes, more time step restrictions should be considered.

Now construct Finite Element discretizations under LBB condition. In our case, apply P2 Finite Element approximation for the velocity field and P1 Finite Element approximation for the pressure in each element:

$$u_1(x, y, t) = \sum_{j=1}^6 U_{1j} \psi_j(x, y) \quad (3.54)$$

$$u_2(x, y, t) = \sum_{j=1}^6 U_{2j} \psi_j(x, y) \quad (3.55)$$

$$p(x, y, t) = \sum_{j=1}^3 P_j \lambda_j(x, y) \quad (3.56)$$

where ψ and λ are P2 and P1 Finite Element basis functions. m and n are degrees of freedom for the velocity and pressure field, and U_{1j} , U_{2j} , P_j , are nodal values for velocity and pressure as before. Follow the notations used in (3.4) and (3.5), we denote the Finite Element approximation for the test function and velocity field as follows:

$$\mathcal{V}^h = \{w^h \in H^1(\Omega) | w^h|_K \in \mathcal{P}_2(k) \quad \forall K \in \mathcal{T}_h \quad \text{and} \quad w^h = 0 \quad \text{on} \quad \Sigma_D\} \quad (3.57)$$

$$\mathcal{S}_i^h = \{u_i^h \in H^1(\Omega) | u_i^h|_K \in \mathcal{P}_2(k) \quad \forall K \in \mathcal{T}_h \quad \text{and} \quad u_i^h = b_i^h \quad \text{on} \quad \Sigma_D\} \quad (3.58)$$

where $i = 1, 2$. The resulting Finite Element discretization for this convection diffusion step comes up with the following algebraic equations after assembling:

$$\begin{aligned}
& [M^n + \theta\Delta t C^n + \theta\Delta t(K^n + K_{11}^n)]U_1^{n+1} + \theta\Delta t K_{21}^n U_2^{n+1} \\
& = [M^n - (1 - \theta)\Delta t C^n - (1 - \theta)\Delta t(K^n + K_{11}^n)]U_1^n - (1 - \theta)\Delta t K_{21}^n U_2^n \\
& \quad + \theta\Delta t G_1^{n+1} + (1 - \theta)\Delta t G_1^n \\
& \quad + \Delta t I_1
\end{aligned} \tag{3.59}$$

$$\begin{aligned}
& [M^n + \theta\Delta t C^n + \theta\Delta t(K^n + K_{22}^n)]U_2^{n+1} + \theta\Delta t K_{12}^n U_1^{n+1} \\
& = [M^n - (1 - \theta)\Delta t C^n - (1 - \theta)\Delta t(K^n + K_{22}^n)]U_2^n - (1 - \theta)\Delta t K_{12}^n U_1^n \\
& \quad + \theta\Delta t G_2^{n+1} + (1 - \theta)\Delta t G_2^n \\
& \quad + \Delta t I_2
\end{aligned} \tag{3.60}$$

From now on we switch to superscript to denote the dependency on ϕ at different time step for conveniences. The element matrices and arrays need to be calculated are summarized in the following table:

Table 3.1: Element matrices and arrays for the first step

M^{j^e}	$\int_{\Omega_e} \rho(\phi^j) \psi \psi^T d\Omega_e$
$C^{j^e}(u, v)$	$\int_{\Omega_e} \rho(\phi^j) [\psi(\psi^T u_1) \frac{\partial \psi^T}{\partial x} + \psi(\psi^T u_2) \frac{\partial \psi^T}{\partial y}] d\Omega_e$
K^{j^e}	$\int_{\Omega_e} \mu(\phi^j) (\frac{\partial \psi}{\partial x} \frac{\partial \psi^T}{\partial x} + \frac{\partial \psi}{\partial y} \frac{\partial \psi^T}{\partial y}) d\Omega_e$
$K_{11}^{j^e}$	$\int_{\Omega_e} \mu(\phi^j) (\frac{\partial \psi}{\partial x} \frac{\partial \psi^T}{\partial x}) d\Omega_e$
$K_{21}^{j^e}$	$\int_{\Omega_e} \mu(\phi^j) (\frac{\partial \psi}{\partial x} \frac{\partial \psi^T}{\partial y}) d\Omega_e$
$K_{12}^{j^e}$	$\int_{\Omega_e} \mu(\phi^j) (\frac{\partial \psi}{\partial y} \frac{\partial \psi^T}{\partial x}) d\Omega_e$
$K_{22}^{j^e}$	$\int_{\Omega_e} \mu(\phi^j) (\frac{\partial \psi}{\partial y} \frac{\partial \psi^T}{\partial y}) d\Omega_e$
$G_1^{j^e}$	$\int_{\Omega_e} \rho(\phi^j) g_1 \psi d\Omega_e$
$G_2^{j^e}$	$\int_{\Omega_e} \rho(\phi^j) g_2 \psi d\Omega_e$

The resulting algebraic system can be solved using standard solvers, for example, GMRES and/or UMFPACK are two good alternatives. How to treat the discontinuous coefficients and approximate the surface tension force as a line integral I are left for discussion on section (3.3) and (3.4.2). The resulting solution from viscous diffusion step is denoted as $\vec{u}^{n+\frac{1}{2}}$, and it is the input for the following pressure projection step.

Step 2. Navier Stokes pressure projection step

Consider the following system

$$\begin{aligned}
 \frac{\partial \vec{u}}{\partial t} + \frac{1}{\rho(\vec{x})} \nabla p &= \vec{0} \quad \text{in } \Omega \times [t_n, t_{n+1}] & (3.61) \\
 \nabla \cdot \vec{u} &= 0 \\
 [\vec{u}]_{\Gamma} \cdot \vec{n} &= \vec{0}, \quad [p]_{\Gamma} = 0 \\
 \vec{u}|_{t=t_n} &= \vec{u}^{n+\frac{1}{2}}
 \end{aligned}$$

we've seen (3.61) result in a Poisson-type equation (3.40). The weak formulation in this case reads:

Find $p^{n+1} \in H^1(\Omega)$, and $p^{n+1}(\vec{x}_0) = 0$, where \vec{x}_0 is any reference point in the computational domain, such that,

$$\Delta t \int_{\Omega} \frac{1}{\rho(\phi^{n+1})} \nabla p^{n+1} \cdot \nabla q d\Omega = - \int_{\Omega} q(\nabla \cdot \vec{u}^{n+\frac{1}{2}}) d\Omega \quad (3.62)$$

for $\forall q \in H^1(\Omega)$. To update u_i^{n+1} at t^{n+1} , the following weak formulation holds:

Find $u_i^{n+1} \in H^1(\Omega)$, $i = 1, 2$, such that,

$$\int_{\Omega} \rho(\phi^{n+1})(u_i^{n+1} - u_i^{n+\frac{1}{2}}) w d\Omega = -\Delta t \int_{\Omega} w \frac{\partial p^{n+1}}{\partial x_i} d\Omega \quad (3.63)$$

for $\forall w \in H^1(\Omega)$.

After Finite Element discretization under LBB condition and assembling, the projection step comes up with the following algebraic systems:

$$\Delta t K_p^{n+1} P^{n+1} = -b \quad (3.64)$$

$$M^{n+1} U_i^{n+1} = M^{n+1} U_i^{n+\frac{1}{2}} - \Delta t P b_i \quad (3.65)$$

where $i = 1, 2$. Notice all matrices in the projection step utilize the Level Set function after convection step as an input. The element matrices and arrays need to be calculated are summarized in the following table:

Table 3.2: Element matrices and arrays for projection step

K_p^{n+1e}	$\int_{\Omega_e} \rho(\psi^{n+1}) \left(\frac{\partial \psi}{\partial x} \frac{\partial \psi}{\partial x}^T + \frac{\partial \psi}{\partial y} \frac{\partial \psi}{\partial y}^T \right) d\Omega_e$
b^e	$\int_{\Omega_e} \psi \left(\frac{\partial \psi^T}{\partial x} u_1^{n+\frac{1}{2}} + \frac{\partial \psi^T}{\partial y} u_2^{n+\frac{1}{2}} \right) d\Omega_e$
Pb_1^e	$\int_{\Omega_e} \psi \frac{\partial \psi^T}{\partial x} p^{n+1} d\Omega_e$
Pb_2^e	$\int_{\Omega_e} \psi \frac{\partial \psi^T}{\partial y} p^{n+1} d\Omega_e$

3.2.3.2 Three-step fractional step projection scheme

The basic idea of three-step fractional step projection scheme is that we solve the convective effect and the viscous diffusion effect separately. So three consecutive problems will be solved: convection equation, viscous diffusion equation and pressure Poisson equation. The numerical solution of previous step serves as the input of the later step. To be specific, we begin with the analysis of the convection equation.

Step 1. Navier Stokes convection step

In this step, we take care of the convective phenomena only. So we want to solve convection equations of the following type:

$$\begin{aligned} \frac{\partial u_i}{\partial t} + \vec{u} \cdot \nabla u_i &= 0 \quad \text{in } \Omega \times [t_n, t_{n+1}] \\ u_i|_{t=t_n} &= u_i^{(n)} \end{aligned} \quad (3.66)$$

where $\vec{u} = (u_1, u_1)^T$, $i = 1, 2$, with corresponding boundary conditions. Due to the convective property of these equations, we can only prescribe boundary conditions on the inflow part of the boundary. In this case, with the partition of the boundary $\Sigma = \overline{\Sigma_{in}} \cup \overline{\Sigma_{out}}$, where $\Sigma_{in} = \{\vec{x} \in \Sigma | \vec{u} \cdot \vec{n} < 0\}$ and $\Sigma_{out} = \{\vec{x} \in \Sigma | \vec{u} \cdot \vec{n} > 0\}$, \vec{n} is the outward normal of Σ , the Dirichlet boundary condition (3.31) $u_i = b_i$, $i = 1, 2$, can be only prescribed on Σ_{in}

part of the boundary.

The nonlinear term $\vec{u} \cdot \nabla u_i$ in the system again needs more attention. Especially in the context of Finite Element approach for transient problem, we address here again, unlike linear spatial operator with constant coefficients, that it is better to perform time discretization before the spatial discretization. Besides, stability studies associated with time discretization schemes and linear Finite Elements show that convection problems require higher order temporal accuracy to get reasonable solutions. We present a two-step fourth-order method known as TTG-4A scheme.

The following procedures are performed successively:

$$\tilde{u}_i^n = u_i^n + \frac{1}{3}\Delta t \frac{\partial u_i^n}{\partial t} + \frac{1}{12}(\Delta t)^2 \frac{\partial^2 u_i^n}{\partial t^2} \quad (3.67)$$

$$u_i^* = u_i^n + \Delta t \frac{\partial u_i^n}{\partial t} + \frac{1}{2}(\Delta t)^2 \frac{\partial^2 \tilde{u}_i^n}{\partial t^2} \quad (3.68)$$

It is easy to see this scheme is fourth-order accuracy by plugging (3.67) into (3.68), which is indeed

$$u_i^* = u_i^n + \Delta t \frac{\partial u_i^n}{\partial t} + \frac{1}{2}(\Delta t)^2 \frac{\partial^2 u_i^n}{\partial t^2} + \frac{1}{6}(\Delta t)^3 \frac{\partial^3 u_i^n}{\partial t^3} + \frac{1}{24}(\Delta t)^4 \frac{\partial^4 u_i^n}{\partial t^4} + [O(\Delta t)^5]$$

From the governing equation, we have:

$$\frac{\partial u_i}{\partial t} = -(\vec{u} \cdot \nabla)u_i \quad (3.69)$$

and

$$\begin{aligned}
\frac{\partial^2 u_i}{\partial t^2} &= -\frac{\partial}{\partial t}[(\vec{u} \cdot \nabla)u_i] \\
&= -\frac{\partial \vec{u}}{\partial t} \cdot \nabla u_i - (\vec{u} \cdot \nabla) \frac{\partial u_i}{\partial t} \\
&= (\vec{u} \cdot \nabla) \vec{u} \cdot \nabla u_i + (\vec{u} \cdot \nabla)[(\vec{u} \cdot \nabla)u_i]
\end{aligned} \tag{3.70}$$

Plug (3.69) and (3.70) back into (3.67) and (3.68), which gives:

$$\tilde{u}_i^n = u_i^n - \frac{1}{3}\Delta t(\vec{u}^n \cdot \nabla)u_i^n + \frac{1}{12}(\Delta t)^2(\vec{u}^n \cdot \nabla)\vec{u}^n \cdot \nabla u_i^n + \frac{1}{12}(\Delta t)^2(\vec{u}^n \cdot \nabla)[(\vec{u}^n \cdot \nabla)u_i^n] \tag{3.71}$$

$$u_i^* = u_i^n - \Delta t(\vec{u}^n \cdot \nabla)u_i^n + \frac{1}{2}(\Delta t)^2(\vec{u}^n \cdot \nabla)\vec{u}^n \cdot \nabla u_i^n + \frac{1}{2}(\Delta t)^2(\vec{u}^n \cdot \nabla)[(\vec{u}^n \cdot \nabla)u_i^n] \tag{3.72}$$

Now follow FEM mentioned in section (3.1), the weak formulations for (3.67) and (3.68) read:

Find $\tilde{u}_i^n \in H^1(\Omega)$ and $\tilde{u}_i^n = b_i$ on Σ_{in} , $i = 1, 2$, such that

$$\begin{aligned}
\int_{\Omega} \tilde{u}_i^n w d\Omega &= \int_{\Omega} u_i^n w d\Omega - \frac{1}{3}\Delta t \int_{\Omega} \vec{u}^n \cdot \nabla u_i^n w d\Omega \\
&\quad + \frac{1}{12}(\Delta t)^2 \int_{\Omega} \vec{u}^n \cdot \nabla \vec{u}^n \cdot \nabla u_i^n w d\Omega \\
&\quad + \frac{1}{12}(\Delta t)^2 \int_{\Sigma_{out}} (\vec{u}^n \cdot \nabla u_i^n)(w \vec{u}^n \cdot \vec{n}) d\Sigma \\
&\quad - \frac{1}{12}(\Delta t)^2 \int_{\Omega} (\vec{u}^n \cdot \nabla u_i^n)(\nabla w \cdot \vec{u}^n) d\Omega \\
&\quad - \frac{1}{12}(\Delta t)^2 \int_{\Omega} (\vec{u}^n \cdot \nabla u_i^n)(w \nabla \cdot \vec{u}^n) d\Omega
\end{aligned} \tag{3.73}$$

for $\forall w \in H^1(\Omega)$ and $w = 0$ on Σ_{in} .

Find $u_i^* \in H^1(\Omega)$ and $u_i^* = b_i$ on Σ_{in} , $i = 1, 2$, such that

$$\begin{aligned}
\int_{\Omega} u_i^* w d\Omega &= \int_{\Omega} u_i^n w d\Omega - \Delta t \int_{\Omega} \tilde{u}^n \cdot \nabla u_i^n w d\Omega \\
&+ \frac{1}{2} (\Delta t)^2 \int_{\Omega} \tilde{\tilde{u}}^n \cdot \nabla \tilde{\tilde{u}}^n \cdot \nabla \tilde{u}_i^n w d\Omega \\
&+ \frac{1}{2} (\Delta t)^2 \int_{\Sigma_{out}} (\tilde{\tilde{u}}^n \cdot \nabla \tilde{u}_i^n) (w \tilde{\tilde{u}}^n \cdot \vec{n}) d\Sigma \\
&- \frac{1}{2} (\Delta t)^2 \int_{\Omega} (\tilde{\tilde{u}}^n \cdot \nabla \tilde{u}_i^n) (\nabla w \cdot \tilde{\tilde{u}}^n) d\Omega \\
&- \frac{1}{2} (\Delta t)^2 \int_{\Omega} (\tilde{\tilde{u}}^n \cdot \nabla \tilde{u}_i^n) (w \nabla \cdot \tilde{\tilde{u}}^n) d\Omega
\end{aligned} \tag{3.74}$$

for $\forall w \in H^1(\Omega)$ and $w = 0$ on Σ_{in} .

Remark 3.11. *The scheme is explicit, and the nonlinear effect is not a problem due to this explicit character. Besides, this scheme is efficient since two components of the velocity field can be sought separately. After Finite Element approximation, we will come up with two similar algebraic systems with exact same mass matrix for each step. The mass matrix has been calculated in (3.15), so we do not have to reassemble it over the loop.*

Remark 3.12. *Courant, Friedrichs, and Lewy [27] pointed out that the time step can not be chosen arbitrarily for this hyperbolic type of PDE. The scheme is stable if the following Courant-Friedrichs-Lewy (CFL) condition holds*

$$C = \frac{|\vec{u}| \Delta t}{h} < C_{max} \tag{3.75}$$

where usually $C_{max} = 1$ for the explicit scheme. C_{max} might be larger when the implicit scheme is applied. The CFL condition determines the interval time length for this Navier Stokes convection step. Later in the Level Set convection step, this CFL condition still has

to be satisfied.

Remark 3.13. After a simple algebraic modification of the weak formulation of (3.73), we have

$$\begin{aligned}
\int_{\Omega} \frac{\tilde{u}_i^n - u_i^n}{\Delta t} w d\Omega &= -\frac{1}{3} \int_{\Omega} \bar{u}^n \cdot \nabla u_i^n w d\Omega \\
&+ \frac{1}{12} (\Delta t) \int_{\Omega} \bar{u}^n \cdot \nabla \bar{u}^n \cdot \nabla u_i^n w d\Omega \\
&+ \frac{1}{12} (\Delta t) \int_{\Sigma_{out}} (\bar{u}^n \cdot \nabla u_i^n) (w \bar{u}^n \cdot \bar{n}) d\Sigma \\
&- \frac{1}{12} (\Delta t) \int_{\Omega} (\bar{u}^n \cdot \nabla u_i^n) (\nabla w \cdot \bar{u}^n) d\Omega \\
&- \frac{1}{12} (\Delta t) \int_{\Omega} (\bar{u}^n \cdot \nabla u_i^n) (w \nabla \cdot \bar{u}^n) d\Omega
\end{aligned}$$

It is interesting to see that all terms on the right-hand side have $(\bar{u}^n \cdot \nabla u_i^n)$. They can be comprehended as the linearization of the nonlinear convection term in the governing equation with a uniform velocity \bar{u}^n . The surface integral $\int_{\Sigma_{out}} (\bar{u}^n \cdot \nabla u_i^n) (w \bar{u}^n \cdot \bar{n}) d\Sigma$ is evaluated only at the outflow part of the boundary, because the test function is zero on the inflow part of the boundary. This boundary term takes into account the outflow of fluid momentum along the streamlines of the advection field, and it guarantees good absorbing properties at outflow boundaries in transient calculations. The weak formulation for the second step in Taylor Galerkin method can be interpreted in the same fashion.

The resulting Finite Element discretization for this convection step come up with the following algebraic equations on each element:

$$M^e(\tilde{U}_i^n - U_i^n) = \frac{1}{3}\Delta t C_1^e U_i^n + \frac{1}{12}(\Delta t)^2 C_2^e U_i^n \quad (3.76)$$

$$M^e(U_i^* - U_i^n) = \Delta t C_1^e U_i^n + \frac{1}{2}(\Delta t)^2 C_2^e \tilde{U}_i^n \quad (3.77)$$

where $i = 1, 2$ and the matrix M^e is the standard consistent mass matrix mentioned on section (3.1) in (3.15). After assembling, the following algebraic systems should be solved in the convection step:

$$M(\tilde{U}_i^n - U_i^n) = \frac{1}{3}\Delta t C_1 U_i^n + \frac{1}{12}(\Delta t)^2 C_2 U_i^n \quad (3.78)$$

$$M(U_i^* - U_i^n) = \Delta t C_1 U_i^n + \frac{1}{2}(\Delta t)^2 C_2 \tilde{U}_i^n \quad (3.79)$$

There is no density term in the convective step of the three-step scheme, so discontinuous integrations are not needed for this step.

The resulting solution from convection equation is denoted as $\vec{u}^{n+1} = \vec{u}^*$, and it is the initial condition for the following viscous diffusion step.

Step 2. Navier Stokes viscous diffusion step

In this step, we consider the viscous diffusion process as well as the surface tension force in the following viscous diffusion system:

$$\rho(\vec{x}) \frac{\partial \vec{u}}{\partial t} - \nabla \cdot (\mu(\vec{x})(\nabla \vec{u} + \nabla \vec{u}^T)) = \rho(\vec{x}) \vec{g} \quad \text{in } \Omega \times [t_n, t_{n+1}] \quad (3.80)$$

$$[\vec{u}]_{\Gamma} = \vec{0}$$

$$-[\mu(\vec{x})(\nabla \vec{u} + \nabla \vec{u}^T)]_{\Gamma} \cdot \vec{n} = \sigma \kappa \vec{n}$$

$$\vec{u}|_{t=t_n} = \vec{u}^*$$

with corresponding boundary conditions. Unlike the convection step, we have to specify boundary conditions on all boundaries Σ of our computational domain. Perform spatial discretization first. We have the following:

Find $\vec{u} \in H^1(\Omega) \times H^1(\Omega)$ and $\vec{u} = \vec{b}$ on Σ , such that,

$$\int_{\Omega} \rho(\phi) \frac{\partial \vec{u}}{\partial t} \cdot \vec{w} d\Omega + \int_{\Omega} \mu(\phi) (\nabla \vec{u} + \nabla \vec{u}^T) : \nabla \vec{w}^T d\Omega = \int_{\Omega} \rho(\phi) \vec{g} \cdot \vec{w} d\Omega + \int_{\Gamma} \kappa \sigma \vec{n} \cdot \vec{w} \quad (3.81)$$

for $\forall \vec{w} \in H^1(\Omega) \times H^1(\Omega)$ such that $\vec{w} = \vec{0}$ on Σ .

Follow the previous work of two-step method, this diffusion equation can be solved in the linearized and/or non-linearized version in the same fashion without the convection term.

Step 3. Navier Stokes pressure projection step

Navier Stokes pressure projection step follows exact the same idea as in the second step of Chorin Teman projection scheme. The intermediate velocity field calculated from the viscous diffusion step is used as input. A Poisson-type equation is solved, and the velocity field is recovered by the updated pressure field in the projection step.

3.2.3.3 Van Kan's projection scheme

For the Van Kan's projection scheme, we need to insert the old pressure gradient into the momentum equation and solve the viscous Burgers' equation directly in the first step. Then we need to project the resulting velocity field onto the subspace of solenoidal functions to update the pressure field. We illustrate the linearized version concisely. To be specific, the following two steps are performed successively,

Step one of linearized Van Kan scheme

Linearized Van Kan Scheme is similar to linearized semi-discrete/two-step version, whereas the pressure term evaluated at previous time step should be included on the RHS of (3.46), namely, the following system has to be solved first:

$$\begin{aligned}
 (M_n + \theta\Delta t(C_n + K_n))U^{n+\frac{1}{2}} &= (M_n - (1 - \theta)\Delta t(C_n + K_n))U^n & (3.82) \\
 &- \Delta t B^T P^n \\
 &+ \theta\Delta t b^{n+\frac{1}{2}} + (1 - \theta)\Delta t b^n
 \end{aligned}$$

Notice this step is basically the same as the first step in Chorin Teman scheme. The only difference is the presence of gradient of pressure term on the right-hand side of the weak formulation. All other terms are exactly the same as those in the first step of Chorin Teman scheme. As we shall see in the static bubble test, this presence of gradient pressure term does not give good behavior when surface tension force is included.

Step two of linearized Van Kan Scheme

The second step of the Van Kan scheme is slightly different from two previous schemes by introducing pressure difference $q^{n+1} = \frac{p^{n+1} - p^n}{2}$. The pressure difference makes this scheme second order. However, a Poisson-type equation is also obtained. The following two steps complete Van Kan scheme:

Find $q^{n+1} \in H^1(\Omega)$, and $q^{n+1}(\vec{x}_0) = 0$, where \vec{x}_0 is any reference point in the computational domain, such that,

$$\Delta t \int_{\Omega} \frac{1}{\rho(\phi^{n+1})} \nabla q^{n+1} \cdot \nabla v d\Omega = - \int_{\Omega} v (\nabla \cdot \vec{u}^{n+\frac{1}{2}}) d\Omega \quad (3.83)$$

for $\forall v \in H^1(\Omega)$. To update u_i^{n+1} at t^{n+1} , the following weak formulation holds:

Find $u_i^{n+1} \in H^1(\Omega)$, $i = 1, 2$, such that,

$$\int_{\Omega} \rho(\phi^{n+1})(u_i^{n+1} - u_i^{n+\frac{1}{2}})w d\Omega = -\Delta t \int_{\Omega} w \frac{\partial q^{n+1}}{\partial x_i} d\Omega \quad (3.84)$$

for $\forall w \in H^1(\Omega)$.

As we have seen in this section, the Van Kan's projection scheme follows similar ideas as the three-step fractional step projection method. But it only consists of two steps: in the first step a viscous Burgers equation is solved with the presence of old pressure gradient in the momentum equation. In the second step the incompressibility constraint is considered by projecting the resulting velocity field from the first step on to the subspace of solenoidal functions, which also leads to a Poisson-type equation. Basically, Van Kan's projection scheme combines the convection step and the viscous diffusion step from the three-step projection method with the presence of the gradient pressure. As a projection scheme, Van Kan's scheme also suffers from wrong boundary conditions in the projection step, and it can give a spurious pressure boundary layer with width $O(\sqrt{\nu\Delta t})$.

3.3 Numerical integration of discontinuous functions

As mentioned in the previous section, evaluating discontinuous densities and viscosities precisely is one of our major numerical challenges. The densities and viscosities vary across the interface. Mathematically, discontinuous densities and viscosities can be expressed via the following Characteristic function:

Introducing a Characteristic function as an indicator of fluid 1 as follows,

$$\chi_1(\vec{x}) = \begin{cases} 1 & \vec{x} \text{ in } \Omega_1 \\ 0 & \vec{x} \text{ in } \Omega_2 \end{cases} \quad (3.85)$$

the discontinuous densities and viscosities can be expressed as:

$$\rho(\vec{x}) = \rho_2 + (\rho_1 - \rho_2)\chi_1(\vec{x}) \quad (3.86)$$

$$\mu(\vec{x}) = \mu_2 + (\mu_1 - \mu_2)\chi_1(\vec{x}) \quad (3.87)$$

In general, we want to evaluate integrations of the following form:

$$\int_{\Omega} \chi f d\Omega \quad (3.88)$$

where f is an arbitrary smooth function. As we have mentioned in the previous chapters, how to evaluate this type of integral accurately is essentially important in our two-phase flow simulation. In addition, the weak formulation of the equations also includes the integral of the surface tension force over the interface. In FEM, the line integration becomes

$$\int_{\Gamma} \vec{f} \cdot \vec{w} d\Gamma = \int_{\Omega} \delta(\Gamma) \vec{f} \cdot \vec{w} d\Omega \quad (3.89)$$

This kind of integral involves approximation of the singular function with support on the interface. Evaluating this type of integral is very important especially when surface tension presents. Later in the section of approximation of surface tension force, we shall see this type of integration can be evaluated without approximating the singular function when a semi-implicit scheme is adopted. Although this involves localizing the zero contours of the interface, we need to calculate the areas inside the contours anyway in the proposed

Volume Fraction method. From now on, we only focus on the numerical integration of the discontinuous type. There are two popular approaches to approximate this type of integral traditionally. The first one requires to keep the Characteristic function and modify numerical quadratures accordingly. This method yields an $O(h^2)$ error when the interface is approximated by a straight line in each interfacial triangle, see [57]. The second method involves expressing the Characteristic function χ as a Heaviside function of the following type:

$$\chi(x) = H(\phi(x)) \quad (3.90)$$

where

$$H(t) = \begin{cases} 0 & \text{for } t < 0 \\ \frac{1}{2} & \text{for } t = 0 \\ 1 & \text{for } t > 0 \end{cases} \quad (3.91)$$

To regularized this Heaviside function, the following regularized functions are applied:

$$H_\varepsilon(t) = \begin{cases} 0 & \text{for } t < -\varepsilon \\ \nu(t/\varepsilon) & \text{for } |t| < \varepsilon \\ 1 & \text{for } t > \varepsilon \end{cases} \quad (3.92)$$

where $\nu(\xi)$ is a smooth transition function with $\nu(-1) = 0$ and $\nu(1) = 1$. $\nu(\xi) = \frac{1}{2}(1 + \xi + \frac{1}{\pi}\sin(\pi\xi))$ is a popular choice. Errors introduced by this approach are also analyzed in [57].

Here, we introduce a new Volume Fraction method to calculate discontinuous densities and viscosities. The idea is that without regularizing the Characteristic function, the Volume Fraction for indicated fluid is calculated first. Then we evaluate the discontinuous

integrals using the resulting Volume Fraction function.

3.3.1 Volume Fraction method

Here, we propose the following Volume Fraction method. We use the Characteristic function as an indicator of fluid one like before:

$$\chi_1(\vec{x}) = \begin{cases} 1 & \vec{x} \text{ in } \Omega_1 \\ 0 & \vec{x} \text{ in } \Omega_2 \end{cases} \quad (3.93)$$

The Volume Fraction on degree of freedom i is defined using the Characteristic function χ_1 as follows:

$$\alpha_i = \frac{\int_{\text{area occupied by fluid 1}} \chi_1 d\Omega}{\int_{\text{area of triangles containing } i} 1 d\Omega} \quad (3.94)$$

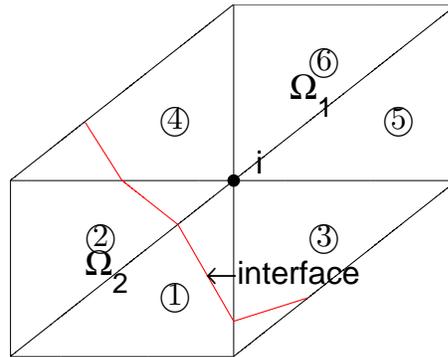


Figure 3.4: Volume Fraction method

As shown in figure (3.4), α_i is calculated as the fraction of area occupied by fluid one and the area of six triangles containing i . It is clear that $\alpha_i \in [0, 1]$. Recall the area of an arbitrary triangle with vertices (x_1, y_1) , (x_2, y_2) , (x_3, y_3) can be evaluated using the

following determinant:

$$\text{area of triangle} = \frac{1}{2} \begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix}. \quad (3.95)$$

The numerator in (3.94) can be calculated as long as the interface is located. In each interfacial element, for example in Figure (3.5), interface can be located with the sign of Level Set function ϕ evaluated on each of degree of freedom. As $\phi|_{(x_3, y_3)} < 0$, $\phi|_{(x_1, y_1)} > 0$, and $\phi|_{(x_2, y_2)} > 0$, i.e. $(x_3, y_3) \in \Omega_1$, and $(x_1, y_1), (x_2, y_2) \in \Omega_2$, the interface can be approximated with segment connecting points (x_a, y_a) , and (x_b, y_b) . The coordinates of points (x_a, y_a) , and (x_b, y_b) are obtained via the following interpolation:

$$\begin{cases} x_a = x_1 + (x_3 - x_1) \frac{\phi_1}{(\phi_1 - \phi_3)} \\ y_a = y_1 + (y_3 - y_1) \frac{\phi_1}{(\phi_1 - \phi_3)} \end{cases} \quad (3.96)$$

$$\begin{cases} x_b = x_2 + (x_3 - x_2) \frac{\phi_2}{(\phi_2 - \phi_3)} \\ y_b = y_2 + (y_3 - y_2) \frac{\phi_2}{(\phi_2 - \phi_3)} \end{cases} \quad (3.97)$$

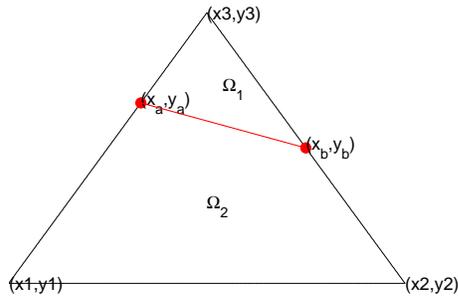


Figure 3.5: Interfacial element

Now in each element, define function $\alpha_h(\vec{x})$ as a linear combination of P1 finite element

basis functions λ_k , $k = 1, 2, 3$, with Volume Fraction as weight, namely,

$$\alpha_h^e(\vec{x}) = \lambda^T \alpha \quad (3.98)$$

where $\alpha = (\alpha_1, \alpha_2, \alpha_3)^T$ is the Volume Fraction, and $\lambda = (\lambda_1, \lambda_2, \lambda_3)^T$ is the P1 Finite Element basis function. The discontinuous densities and viscosities are approximated using $\alpha_h^e(\vec{x})$:

$$\rho_h^e(\vec{x}) = \alpha_h^e(\vec{x})\rho_1 + (1 - \alpha_h^e(\vec{x}))\rho_2 \quad (3.99)$$

$$\mu_h^e(\vec{x}) = \alpha_h^e(\vec{x})\mu_1 + (1 - \alpha_h^e(\vec{x}))\mu_2 \quad (3.100)$$

Follow the notations in section (3.1), the mass matrix in viscous diffusion step M^{i^e} is calculated in the following way:

$$\begin{aligned} M^{i^e} &= \int_{\Omega_e} \rho(\phi^i) \psi \psi^T d\Omega_e \\ &= \int_{\Omega_e} (\alpha_h^{i^e} \rho_1 + (1 - \alpha_h^{i^e}) \rho_2) \psi \psi^T d\Omega_e \\ &= \rho_2 \int_{\Omega_e} \psi \psi^T d\Omega_e + (\rho_1 - \rho_2) \int_{\Omega_e} \alpha_h^{i^e} \psi \psi^T d\Omega_e \\ &= \rho_2 \int_{\Omega_e} \psi \psi^T d\Omega_e + (\rho_1 - \rho_2) \int_{\Omega_e} (\lambda^T \alpha) \psi \psi^T d\Omega_e \end{aligned} \quad (3.101)$$

The first integral is the standard mass matrix calculated in (3.15), and the second integral can be calculated in the similar way as mentioned in section (3.1), which is,

$$\int_{\Omega_e} (\lambda^T \alpha) \psi \psi^T d\Omega_e = A \int_{\Omega_e} (\lambda^T \alpha) R R^T d\Omega_e A^T \quad (3.102)$$

To calculate mass matrix with discontinuous coefficients, we only need to update the Volume Fractions, namely α_s . The integrals are exact integrations given in proposition (3.3). Besides, other matrices and arrays with discontinuous coefficients can be calculated in the same manner.

3.3.2 Numerical test cases and error analysis of Volume Fraction method

In the numerical tests, we perform grid convergence experiments to study error behaviors of Volume Fraction method. Let the computational domain be a unit square $\Omega = [0, 1] \times [0, 1]$, with a fixed square interface Ω_s . The fixed interface has four corners $(2/5, 2/3)$, $(3/5, 2/3)$, $(2/5, 13/15)$, and $(3/5, 13/15)$, see Figure (3.6). The area of this fixed square is simply 0.04. Convergence tests are performed on isosceles right triangle meshes with interval size h . The longest side of each triangle is $\sqrt{2}h$. Refinement is performed such that after refinement, the interval size becomes $h/2^l$, for $l = 1, 2, 3, \dots, 7$, and the longest side of each triangle after refinement becomes $\sqrt{2}h/2^l$.

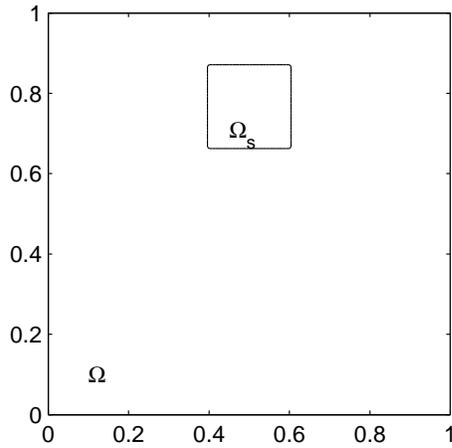


Figure 3.6: Computational domain for discontinuous integrals

Four different discontinuous functions are chosen as the integrates. For the first example, we take $f = 1$. We want to integrate a Characteristic function defined as one if it is located inside the little square Ω_s , and zero otherwise, which is:

$$\chi(\vec{x}) = \begin{cases} 1 & \vec{x} & \text{in} & \Omega_s \\ 0 & \vec{x} & \text{in} & \Omega \setminus \Omega_s \end{cases} \quad (3.103)$$

The integration of χ over Ω is the area of Ω_s . The Figure (3.7) shows the loglog plot of the error over mesh sizes, where error is defined as follows:

$$error = \left| \int_{\Omega} \chi d\Omega - \int_{\Omega} \alpha d\Omega \right|$$

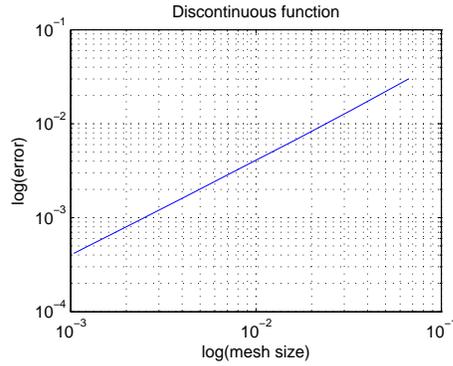


Figure 3.7: Error of integration of Characteristic function

An alternative approach to calculate the Volume Fraction is to use L2 projection of the Characteristic function, namely

$$(\alpha, \lambda) = (\chi, \lambda) \quad (3.104)$$

for all test functions λ . (\cdot, \cdot) denotes the L2 scalar product. If the Volume Fraction is

calculated this way, the error above will be zero.

More generally, we want to evaluate the numerical integration of discontinuous functions with the following form:

$$\int_{\Omega} \chi f d\Omega$$

Actually, $\int_{\Omega} \chi f d\Omega = \int_{\Omega_s} f d\Omega_s$. We want to investigate error behaviors over refinement, and the error is defined as:

$$error = \left| \int_{\Omega} \chi f d\Omega - \int_{\Omega} \alpha f d\Omega \right|$$

The second and third tests involve integrations of second and third order polynomials, where $f = x^2 + y^2$, and $f = x^4 + y^4$ respectively. The exact integration of these kinds of polynomial can be easily evaluated, and loglog plot of the error over mesh sizes are given in the following figures:

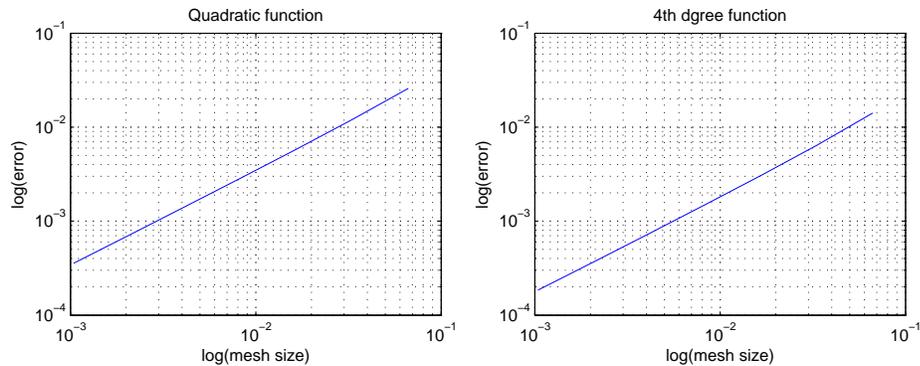


Figure 3.8: Error of integration of second order and fourth order discontinuous functions

The slopes in loglog plot in the first three test cases are 1.0249, 1.0275, and 1.0382 respectively. So our scheme is order one.

The last numerical test is performed with $f = \exp(\sqrt{x^2 + y^2})$. Since no explicit integration formula can be found in this case, the following table shows the convergence rate of our Volume Fraction scheme:

Table 3.3: numerical integration of $\int_{\Omega} \chi \exp(\sqrt{x^2 + y^2}) d\Omega$

h	u_h	$u_h - u_{h/2}$	$\frac{u_h - u_{h/2}}{u_{h/2} - u_{h/4}}$	$\log_2\left(\frac{u_h - u_{h/2}}{u_{h/2} - u_{h/4}}\right)$
1./15	0.176153	0.040182	2.182618142	1.1261
1./30	0.135971	0.01841	2.093949045	1.0662
1./60	0.117561	0.008792	2.047508151	1.0339
1./120	0.108769	0.004294	2.024516737	1.0176
1./240	0.104475	0.002121	2.012333966	1.0089
1./480	0.102354	0.001054	2.003802281	1.0027
1./960	0.1013			

With all these tests, we conclude that our Volume Fraction method is first order accuracy. We will see it is a stable method when evaluating the discontinuous coefficients in the rising bubble benchmark problem in Chapter 4.

3.4 Numerical representation of the interface

3.4.1 Approximation of Level Set function

As prescribed in section (2.2.2), we need to solve the following type of PDE for the interface

$$\frac{\partial \phi}{\partial t} + \vec{u} \cdot \nabla \phi = 0 \tag{3.105}$$

with initial data. Boundary conditions are defined only on the inlet of domain. We come up with a convection equation for the Level Set function. This can be solved using TTG-4A, which has been studied in the convection step of three-step method. Two different linear advection equations have been solved using TTG-4A to verify this numerical scheme.

Case One: Linear advection

The computational domain is a rectangle with length $Lx = 10$, and height $Ly = 100$. Initially, a circle is centered at $x_0 = Lx/2$ and $y_0 = Ly - 2R_0$, where $R_0 = 4$ is the radius of the circle. The velocity is taken as $\vec{\mathbf{u}} = (0, -1)^T$. The analytical solution of this test case is: $\phi = R_0 - \sqrt{(x - x_0)^2 + (y - y_0 - time)^2}$. Figure (3.9) shows the computational results for the interface ϕ at different times using TTG-4A.

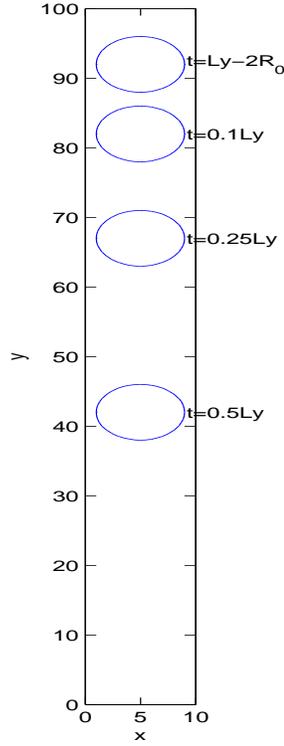


Figure 3.9: Interface position of linear advection test with mesh size $h = 1./20$, time step $\Delta t = 0.25h$ using TTG-4A. The problem was solved in non-dimensional variables, see Remark 3.14.

Remark 3.14. *The primary goal of this dissertation is to study and develop robust algorithms to solve the two-phase flow problem, so all parameters and computed numerical quantities including the dimensions, density, viscosity, time, velocity, pressure, force and interface are dimensionless.*

We define relative errors and absolute errors respectively as follows:

$$\begin{aligned}
reErr_1 &= \frac{\int |\phi - \phi_e| d\Omega}{\int |\phi| d\Omega} \\
reErr_2 &= \sqrt{\frac{\int |\phi - \phi_e|^2 d\Omega}{\int \phi^2 d\Omega}} \\
Err_1 &= \int |\phi - \phi_e| d\Omega \\
Err_2 &= \sqrt{\int |\phi - \phi_e|^2 d\Omega}
\end{aligned}$$

The error behaviors versus refinement levels are presented in Figure (3.10).

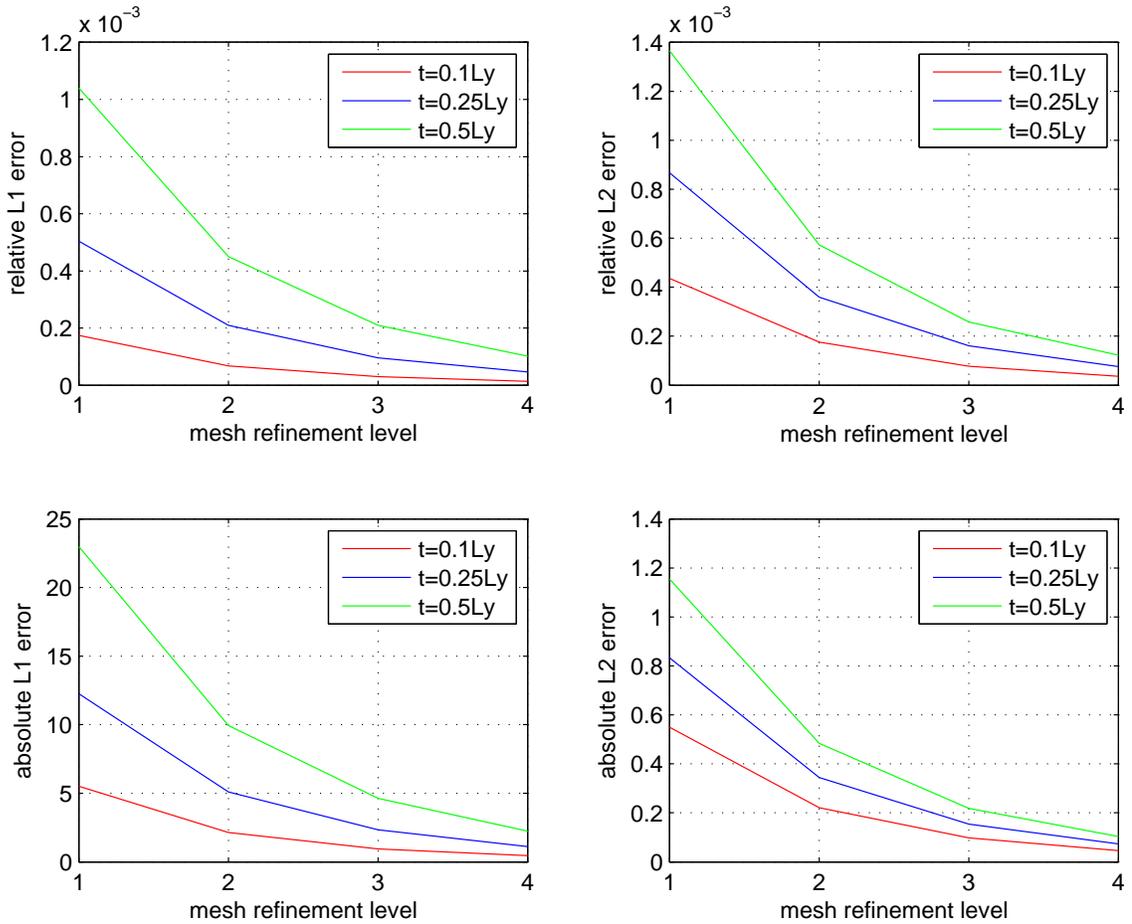


Figure 3.10: Errors for 2D linear advection test under grid refinement. The problem was solved in non-dimensional variables, see Remark 3.14.

Case Two: Zalesak’s rotating disc

The computational domain is a square with $Lx = Ly = 1$. Initially, a slotted disc is centered at $(x_0, y_0) = (\frac{1}{2}Lx, \frac{3}{4}Ly)$, with radius $R_0 = \frac{3}{20}Lx$, and $w = \frac{1}{3}R_0$. Here, the advection velocity is taken as $\vec{u} = (-(y - \frac{1}{2}Ly), x - \frac{1}{2}Lx)^T$. After one revolution, the disc supposed to rotate back to its initial position. The followings are grid convergence results obtained using TTG-4A after one revolution. The red curves are the computed results, and the blue curves are the initial positions. The problem was solved in non-dimensional variables, see

Remark 3.14.

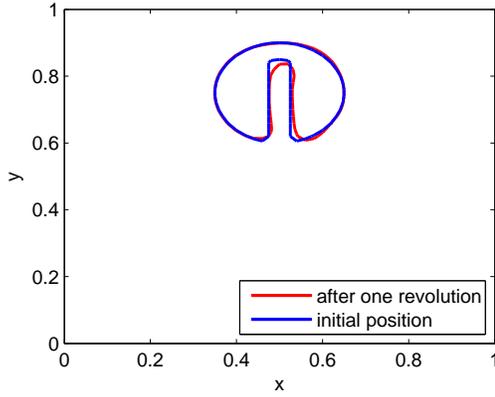


Figure 3.11: Mesh 50×50

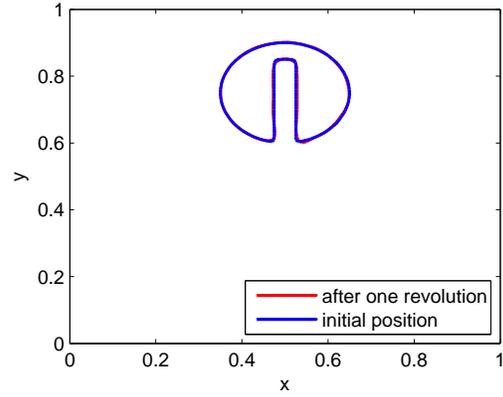


Figure 3.12: Mesh 100×100

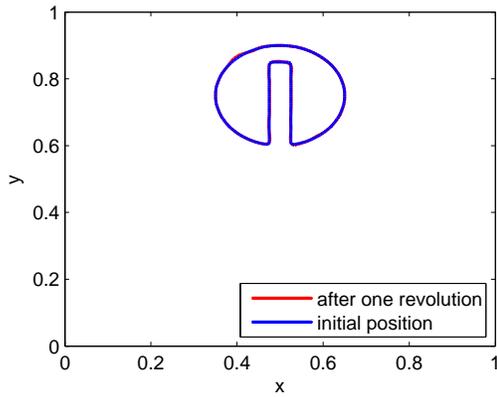


Figure 3.13: Mesh 150×150

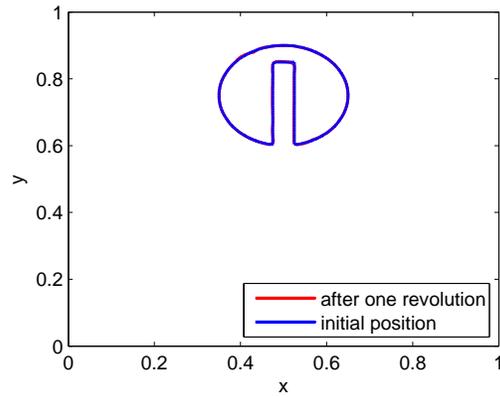


Figure 3.14: mesh 200×200

3.4.2 Approximation of surface tension force

We model the surface tension force mathematically as interfacial boundary conditions. We assume there is no mass transfer between two fluids, i.e. on the interface Γ :

$$[\vec{u}]|_{\Gamma} = \vec{0}, -[-pI + 2\mu(\nabla\vec{u} + \nabla\vec{u}^T)]|_{\Gamma} \cdot \vec{n} = \sigma\kappa\vec{n}$$

where \vec{n} is the interface outward normal, σ is the surface tension coefficient, κ is the curvature of the interface, and $[\cdot]$ denotes the jump property across the interface. We rewrite these boundary conditions as a source term in our two-phase flow problem:

$$\vec{f}_{st} = \sigma \kappa \vec{n} \delta(\Gamma, \vec{x})$$

where the Dirac Delta function $\delta(\Gamma, \vec{x})$ is used here to localize the surface tension force only on the interface between two fluids. As we are using FEM, the surface tension force arises naturally into the weak formulation. In other words, we need to evaluate this type of integral in the weak formulation:

$$\int_{\Omega} \sigma \kappa \vec{n} \cdot \vec{w} \delta(\Gamma, \vec{x}) d\Omega$$

where \vec{w} is the test function. Existence of the Dirac Delta function requires some regularization techniques mentioned in section Numerical integration of discontinuous functions. More detailed analysis of this approach can be found in Tornberg's work, see [59], [57]. Time discretization needs more attention for this force. The most common way is to use the explicit time integration scheme, where all the quantities are evaluated using the interface from the previous time step. So the following integration is included in the right-hand side as a source term:

$$\int_{\Gamma^n} \sigma \kappa^n \vec{n}^n \cdot \vec{w} d\Gamma = \int_{\Omega} \sigma \kappa^n \vec{n}^n \cdot \vec{w} \delta(\Gamma^n, \vec{x}) d\Omega \quad (3.106)$$

Brackbill, Kothe, and Zemach derived the time step criterion for this method in their work [68], which is

$$\Delta t < \sqrt{\frac{\rho h^3}{\sigma}} \quad (3.107)$$

In addition, the curvature κ needs to be evaluated carefully as well if this approach is applied. We follow Hysing's work [19], where a semi-implicit time integration is used. The time step limitation is circumvented. In addition, we do not need to evaluate the curvature term at all.

The semi-implicit method is based on the following results from differential geometry:

$$\underline{\Delta} \vec{x}|_{\Gamma} = \kappa \vec{n}$$

where $\underline{\Delta}$ is the Laplace-Beltrami operator. This operator is defined to be:

$$\underline{\Delta} f(\vec{x}) = \underline{\nabla} \cdot (\underline{\nabla} f(\vec{x})), \quad \vec{x} \in \Gamma$$

If f is two times differentiable in a neighborhood of Γ , the first derivative is given by:

$$\underline{\nabla} f(\vec{x}) = \nabla f(\vec{x}) - (\vec{n}(\vec{x}) \cdot \nabla f(\vec{x})) \vec{n}(\vec{x}), \quad \vec{x} \in \Gamma$$

which is also called tangential gradient of $f(\vec{x})$.

Now we can include the surface tension force in our weak formulations. Multiply the surface tension force with a test function \vec{w} chosen from suitable space, and integrate over Ω , it gives:

$$\begin{aligned}
\vec{f}_\sigma &= \int_{\Omega} \sigma \kappa \vec{n} \cdot \vec{w} \delta(\Gamma, \vec{x}) d\Omega \\
&= \int_{\Gamma} \sigma \kappa \vec{n} \cdot \vec{w} d\Gamma \\
&= \int_{\Gamma} \sigma (\underline{\Delta} \vec{x}|_{\Gamma}) \cdot \vec{w} d\Gamma \\
&= - \int_{\Gamma} \sigma (\underline{\nabla} \vec{x}|_{\Gamma}) \cdot \underline{\nabla} \vec{w} d\Gamma
\end{aligned} \tag{3.108}$$

Bänsch proposed the semi-implicit approach in the context of a Front Tracking method, see [2]. It reduces the awkward time step restriction. This is done by rewriting the new interface position as a function of the old position:

$$(\vec{x}|_{\Gamma})^{n+1} = (\vec{x}|_{\Gamma})^n + \Delta t \vec{u}^{n+1}$$

The integration becomes:

$$\vec{f}_\sigma = - \int_{\Gamma^n} \sigma \underline{\nabla} \vec{x} \cdot \underline{\nabla} \vec{w} ds - \Delta t \int_{\Gamma^n} \sigma \underline{\nabla} \vec{u}^{n+1} \cdot \underline{\nabla} \vec{w} ds \tag{3.109}$$

So we are looking for two integrals for the two components of velocity (σ is constant by assumption):

$$f_{\sigma_1} = -\sigma \int_{\Gamma^n} \underline{\nabla} x \cdot \underline{\nabla} w_1 ds - \sigma \Delta t \int_{\Gamma^n} \underline{\nabla} u_1^{n+1} \cdot \underline{\nabla} w_1 ds \quad (3.110)$$

$$f_{\sigma_2} = -\sigma \int_{\Gamma^n} \underline{\nabla} y \cdot \underline{\nabla} w_2 ds - \sigma \Delta t \int_{\Gamma^n} \underline{\nabla} u_2^{n+1} \cdot \underline{\nabla} w_2 ds \quad (3.111)$$

We propose to do the numerical integration in the following way, so there is no need to approximate the interfacial normals and the curvatures.

We denote the P2 basis functions as ψ_i for $i = 1, 2, \dots, 6$ like before. We can calculate the tangential derivatives as follows:

$$\begin{aligned} \int_{\Gamma^n} \underline{\nabla} x \cdot \underline{\nabla} \psi_i ds &= \frac{x_2 - x_1}{dl} \frac{\psi_i(x_2, y_2) - \psi_i(x_1, y_1)}{dl} dl \\ &= \frac{1}{dl} (x_2 - x_1) (\psi_i(x_2, y_2) - \psi_i(x_1, y_1)) \end{aligned} \quad (3.112)$$

$$\begin{aligned} \int_{\Gamma^n} \underline{\nabla} \psi_j \cdot \underline{\nabla} \psi_i ds &= \frac{\psi_j(x_2, y_2) - \psi_j(x_1, y_1)}{dl} \frac{\psi_i(x_2, y_2) - \psi_i(x_1, y_1)}{dl} dl \\ &= \frac{1}{dl} (\psi_j(x_2, y_2) - \psi_j(x_1, y_1)) (\psi_i(x_2, y_2) - \psi_i(x_1, y_1)) \end{aligned} \quad (3.113)$$

where $dl = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$, and $i, j = 1, 2, \dots, 6$.

As we have seen in the semi-implicit discretization (3.109), the tangential derivative with respect to velocity at time $n + 1$ is a linear term. It represents a diffusion operator, which can be assembled as a positive definite contribution to the iteration matrix on the left-hand side. Instead of approximating the curvatures and the interface normals, tangential derivatives along the interface are calculated. We investigate more of the surface tension effect in the static bubble test in Chapter 4.

Another way to circumventing the time limitation (3.107) is through a fully implicit representation of the interfaces in space. Suppose the Dirac Delta function is constant in the tangential direction:

$$\begin{aligned}
\vec{f}_\sigma &= \int_{\Omega} \sigma \kappa \vec{n} \cdot \vec{w} \delta(\Gamma, \vec{x}) d\Omega \\
&= \int_{\Omega} \sigma (\underline{\Delta} \vec{x}|_{\Gamma}) \cdot (\vec{w} \delta(\Gamma, \vec{x})) d\Omega \\
&= - \int_{\Omega} \sigma (\underline{\nabla} \vec{x}|_{\Gamma}) \cdot \underline{\nabla} (\vec{w} \delta(\Gamma, \vec{x})) d\Omega \\
&= - \int_{\Omega} \sigma (\underline{\nabla} \vec{x}|_{\Gamma}) \cdot \underline{\nabla} \vec{w} \delta(\Gamma, \vec{x}) d\Omega
\end{aligned} \tag{3.114}$$

Apply the semi-implicit time integration again gives:

$$\vec{f}_\sigma = - \int_{\Omega} \sigma \underline{\nabla} \vec{x} \cdot \underline{\nabla} \vec{w} \delta(\Gamma^n, \vec{x}) d\Omega - \Delta t \int_{\Omega} \sigma \underline{\nabla} \vec{u}^{n+1} \cdot \underline{\nabla} \vec{w} \delta(\Gamma^n, \vec{x}) d\Omega \tag{3.115}$$

Compared with the semi-implicit approach, we still have the Dirac Delta functions in these

integrals. Some regularization δ_ϵ has to be constructed, so that the surface tension integration can be implemented fully implicit, and the time step restriction due to the capillary force can be completely removed.

3.4.3 Interface re-initialization

Initially, the Level Set function ϕ is set to be a signed distance function on the interface Γ . The evolution of the Level Set function is governed by the advection equation:

$$\frac{\partial \phi}{\partial t} + \vec{u} \cdot \nabla \phi = 0$$

where \vec{u} is the velocity field of flow determined by solving Navier Stokes equations. The nice property as a signed distance function is generally lost as time goes on. The Level Set function might become too flat or too steep, which makes it difficult to determine the location of the interface. Such oscillations affect the shape of interface as well as the calculations in the surface tension integral, where localization of interface is needed to calculate the tangential derivatives. Therefore, the Level Set function must to be re-initialized, at least after a certain time steps, so that the solution preserves the signed distance property. Given a numerical solution ϕ of the advection equation above, we need to find a solution of the Eikonal type:

$$|\nabla \tilde{\phi}| = 1 \tag{3.116}$$

subject to the interface condition $\tilde{\phi} = 0$ on $\Gamma = \{\vec{x} | \phi(\vec{x}, t) = 0\}$.

Smolianski proposed the use of geometric re-distancing procedures in [52]. Quite a

number of researchers prefer PDE-based techniques, see [37], [9], [60], [39], [49], [69]. Sussman et al. [37] proposed to solve the Eikonal equation by solving the following hyperbolic PDE to the steady state:

$$\begin{aligned} \frac{\partial \tilde{\phi}}{\partial t} + \vec{v} \cdot \nabla \tilde{\phi} &= \text{sign}(\phi) \quad \text{in } \Omega \times (0, t_{steady}) \\ \tilde{\phi}|_{t=0} &= \phi \end{aligned} \tag{3.117}$$

where the velocity is $\vec{v} = \text{sign}(\phi) \frac{\nabla \tilde{\phi}}{|\nabla \tilde{\phi}|}$. Here *sign* is the sign function. Given the Level Set function ϕ obtained from the Level Set convection step initially, a new signed distance function $\tilde{\phi}$ with the same zero Level Set as ϕ is convected with nonlinear transport equation (3.117) to a steady state. It is numerically feasible to solve (3.117) after some time steps such that the signed distance property around the interface is preserved. A regularization of the sign function is needed to obtain a stable solution. For example, the sign function might be replaced by a smooth function $S_\epsilon(\phi)$, where

$$S_\epsilon(\phi) = \frac{\phi}{\sqrt{\phi^2 + \epsilon^2}}, \quad 0 < \epsilon \leq 1$$

Approximation of the sign function brings more numerical diffusions to the transport equation (3.117), and Tornberg [57] showed those numerical diffusions result in inaccuracy of the interface displacement. In addition, the computational cost to solve nonlinear transport equation is very high. Li et al. [13] proposed the distance regularized Level Set evolution algorithm as an alternative approach to realize the re-initialization, where an energy-minimizing gradient flow equation of the following type is solved to the steady state:

$$\frac{\partial \tilde{\phi}}{\partial t} + \frac{\partial R}{\partial \tilde{\phi}} = 0 \quad (3.118)$$

Here $R(\tilde{\phi})$ is a suitable energy functional, and $\frac{\partial R}{\partial \tilde{\phi}}$ is the Euler-Lagrange form of the Gâteaux derivative. For example, if the least squares solution of the Eikonal equation is chosen as the energy functional,

$$R(\tilde{\phi}) = \frac{1}{2} \int_{\Omega} (|\nabla \tilde{\phi}| - 1)^2 d\Omega \quad (3.119)$$

equation (3.118) boils down to the nonlinear heat equation:

$$\begin{aligned} \frac{\partial \tilde{\phi}}{\partial t} - \nabla \cdot \left[\nabla \tilde{\phi} - \frac{\nabla \tilde{\phi}}{|\nabla \tilde{\phi}|} \right] &= 0 \quad \text{in } \Omega \times (0, t_{steady}) \\ \tilde{\phi}|_{t=0} &= \phi \end{aligned} \quad (3.120)$$

Li et al. [13] solved this heat equation by using the Finite Difference method. This method requires severe time step restrictions, and therefore it is still computational expensive. To preserve the same zero contour as the original Level Set function in Finite Element discretization by solving the energy minimizing problem, we apply the recent work of Basting and Kuzmin [21]. The energy functional $R(\tilde{\phi})$ is modified by adding a penalty term of the form:

$$P(\phi, \tilde{\phi}) = \frac{\alpha}{2} \int_{\Gamma(\phi)} \tilde{\phi}^2 ds \quad (3.121)$$

where α is a large positive number used to preserve the shape of the interface. This penalty term gives rise to the following minimization problem:

$$\frac{\partial R}{\partial \tilde{\phi}} + \frac{\partial p}{\partial \tilde{\phi}} = 0 \quad (3.122)$$

It can be solved by the standard Ritz-Galerkin Finite Element approach. With the same choice of functional R in (3.119), the corresponding variational formulation is given by:

$$\int_{\Omega} \left(1 - \frac{1}{|\nabla \tilde{\phi}|}\right) \nabla \tilde{\phi} \cdot \nabla v d\Omega + \alpha \int_{\Gamma(\phi)} \tilde{\phi} v ds = 0 \quad (3.123)$$

for all test functions v . We solve this nonlinear problem using the following fixed point iteration method.

Given $\tilde{\phi}^{(0)} = \phi$, the approximation of correct Level Set function $\tilde{\phi} \approx \tilde{\phi}^{(m)}$ is computed such that

$$a(\tilde{\phi}^{(m)}, v) = b(\tilde{\phi}^{(m-1)}, v) \quad \forall v \quad (3.124)$$

where the bilinear form $a(\cdot, \cdot)$ and linear functional $b(\tilde{\phi}, \cdot)$ are defined by:

$$\begin{aligned} a(\tilde{\phi}, v) &= \int_{\Omega} \nabla \tilde{\phi} \cdot \nabla v d\Omega + \alpha \int_{\Gamma(\phi)} \tilde{\phi} v ds \\ b(\tilde{\phi}, v) &= \int_{\Omega} \frac{\nabla \tilde{\phi}}{|\nabla \tilde{\phi}|} \cdot \nabla v d\Omega \end{aligned} \quad (3.125)$$

The resulting stiffness matrix is independent of $\tilde{\phi}$, and it only needs to be assembled just once outside the iteration loop. Numerical experiments showed a nice signed distance function can be obtained after only a few iterations, see [21]. The advantage of this method is clear: no stabilization or regularization methods are needed, and the re-initialization

approach can preserve the signed distance property in a weak sense.

3.5 Summary of algorithm

Coupling the fluid flows and the modeling of the interface, we summarize our computational strategy in the following:

Algorithm 3 Coupling fluid flows and interface

- 1: Initialization of the Level Set function and velocity (initialization of the pressure field if Van Kan scheme is applied)
 - 2: **for** $t = 1$ to n **do**
 - 3: Level Set convection step
 - 4: Level Set re-initialization step
 - 5: Navier Stokes equations solver
 - 6: **end for**
-

Our first remark to Algorithm 3 is: different approaches can be applied for Navier Stokes equations solver, for example the two-step method, three-step method, and Van Kan scheme. However, the linearized version and non-linearized version have to be carefully investigated when one approximates the discontinuous density and viscosity coefficients; otherwise one will get wrong velocity and pressure field solutions. Another remark is that one can reverse the coupling of fluid flows and interface simulations by solving the Navier Stokes equations first and then solving the Level Set convection and re-initialization accordingly. However, this approach is more time consuming compared with the one we proposed here, since a better approximation of the initial interface is essential for approximation of the discontinuous density and viscosity coefficients in the Navier Stokes equations. This issue has also been mentioned in the work of [16].

Chapter 4

Results

4.1 Advection diffusion equation

Viscous Burgers' equation has a similar structure to the Navier Stokes equations, because it has a nonlinear convection term and a diffusion term with a viscous coefficient. This equation can be considered as a simplified form of the Navier Stokes equations. So we consider the following system of the two-dimensional Burgers' equation:

$$\begin{cases} \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = \frac{1}{R} \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \\ \frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} = \frac{1}{R} \left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) \end{cases}$$

in $\Omega \times [0, T]$, where $\Omega = [0, 1] \times [0, 1]$, $T > 0$. The exact solutions are given by:

$$u(x, y, t) = \frac{3}{4} - \frac{1}{4(1 + \exp(R/32(-t - 4x + 4y)))}; \quad (4.1)$$

$$v(x, y, t) = \frac{3}{4} + \frac{1}{4(1 + \exp(R/32(-t - 4x + 4y)))}; \quad (4.2)$$

where R is the Reynolds number. The numerical tests were performed with $R = 80$. Initial conditions and boundary conditions are obtained from the exact solutions. Three numerical schemes were performed. The first scheme solved the convection and the diffusion term together. The second scheme added a stabilization term to the convection and the diffusion equations. The third scheme solved the convection and diffusion term in two steps. Two components of the velocity field at $t = 1.0$ are obtained with the mesh size 0.025 and the time step $\Delta t = 0.0001$. The numerical solutions at different mesh points along with the exact solutions are reported in the following tables:

Table 4.1: u_1 at $t = 1$

Mesh points	First scheme	Second scheme	Third scheme	Exact solution
(0.1, 0.1)	0.51896	0.518961	0.518962	0.518965
(0.5, 0.1)	0.500425	0.500425	0.500425	0.500375
(0.9, 0.1)	0.500009	0.500009	0.500008	0.500007
(0.3, 0.3)	0.518961	0.518961	0.518962	0.518965
(0.7, 0.3)	0.500425	0.500425	0.500425	0.500375
(0.1, 0.5)	0.704385	0.704384	0.704382	0.7043941
(0.5, 0.5)	0.518961	0.518962	0.518962	0.518965
(0.9, 0.5)	0.500376	0.500376	0.500375	0.500375
(0.1, 0.9)	0.748982	0.748982	0.748982	0.748982
(0.9, 0.9)	0.51896	0.518961	0.518962	0.518965

Table 4.2: u_2 at $t = 1$

Mesh points	First scheme	Second scheme	Third scheme	exact solution
(0.1, 0.1)	0.98104	0.981039	0.981038	0.981035
(0.5, 0.1)	0.999575	0.999575	0.999575	0.999625
(0.9, 0.1)	0.999991	0.999991	0.999992	0.999993
(0.3, 0.3)	0.981039	0.981039	0.981038	0.981035
(0.7, 0.3)	0.999575	0.999575	0.999575	0.999625
(0.1, 0.5)	0.795615	0.795616	0.795618	0.795606
(0.5, 0.5)	0.981039	0.981038	0.981038	0.981035
(0.9, 0.5)	0.999624	0.999624	0.999625	0.999625
(0.1, 0.9)	0.748982	0.751018	0.751018	0.751018
(0.9, 0.9)	0.751018	0.981039	0.981038	0.981035

4.2 Flow around cylinder

Flow around cylinder is a well-documented benchmark problem for one-phase flow. It has been used to study the Computational Fluid Dynamics (CFD) algorithms by many researchers, see [31], [34]. It simulates the fluid flow in a pipe around a fixed circular object. Inflow speeds determine the behavior of the flow. The geometry is set up as follows: a cylinder centered at $(x, y) = (0.2, 0.2)$ with diameter $L = 0.1$ is placed in a channel with length 2.2 and height 0.41, see Figure (4.1).

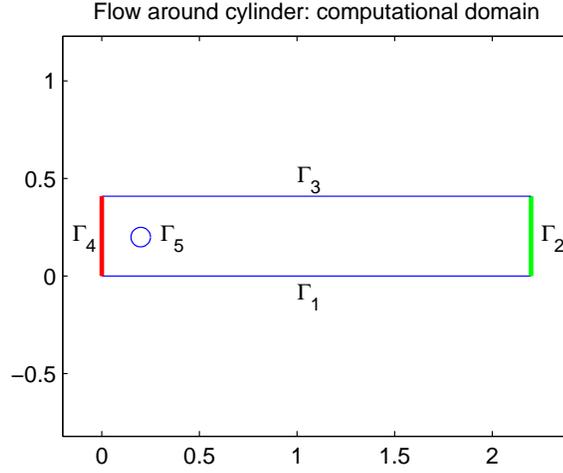


Figure 4.1: Flow around cylinder geometry. The problem was solved in non-dimensional variables, see Remark 3.14.

The fluid is characterized by the stationary Navier Stokes equations with a density $\rho = 1.0$ and a viscosity $\mu = 0.001$. The no-slip Dirichlet boundary conditions are imposed on Γ_1 , Γ_3 , and on the cylinder Γ_5 . Γ_4 is the inflow part, where the following parabolic profile with a maximum inflow velocity $U_{max} = 0.3$ is imposed:

$$(u_1, u_2) = \left(\frac{4U_{max}y(0.41 - y)}{0.41^2}, 0 \right) \quad \text{for all } (x, y) \in [0] \times [0, 0.41]$$

The Neumann boundary conditions are imposed on the outflow part of the boundary Γ_2 .

The corresponding Reynolds number is given as follows:

$$Re = \frac{U_{mean}L}{\mu} = 20$$

Here, $U_{mean} = \frac{2}{3}U_{max} = 0.2$ is the characteristic velocity with the parabolic profile. Our goal is to build up the two-phase flow solver, which simulates the non-stationary fluid flows. However, we can still apply this solver to the stationary problem by simulating the

unsteady problem up to a steady state. The following results are obtained from the steady state, where the speed difference between two time steps is equal to $1e-06$. The speed, the pressure field, and the streamlines are presented as follows with mesh size $h_{max} = 1./60$. Time steps are chosen as $\Delta t = 0.2h_{max}$.

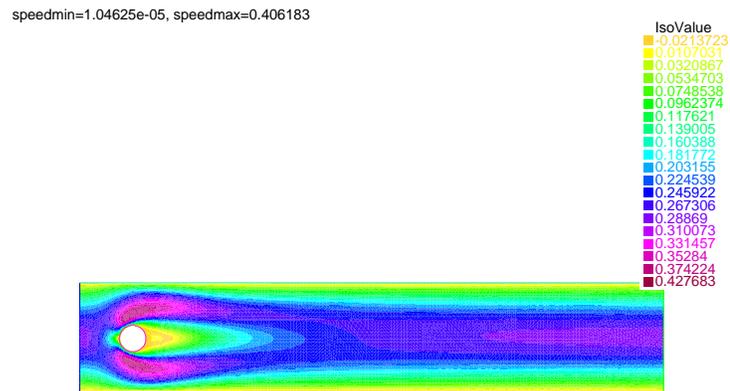


Figure 4.2: Speed of flow around cylinder at the steady state with Reynolds number 20, mesh size $h_{max} = 1./60$, time step $\Delta t = 0.2h_{max}$. The problem was solved in non-dimensional variables, see Remark 3.14.

pmin=-0.0131848, pmax=0.128662

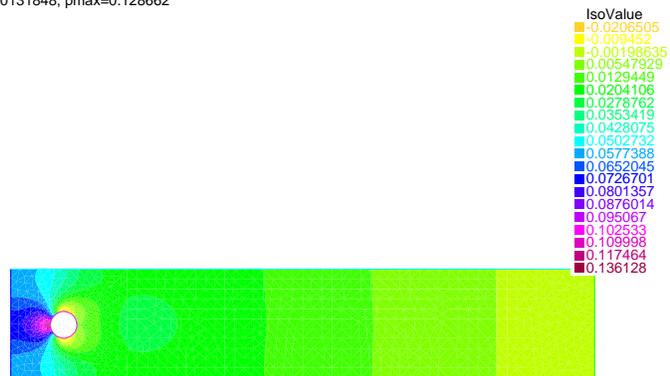


Figure 4.3: Pressure of flow around cylinder at the steady state with Reynolds number 20, mesh size $h_{max} = 1./60$, time step $\Delta t = 0.2h_{max}$. The problem was solved in non-dimensional variables, see Remark 3.14.

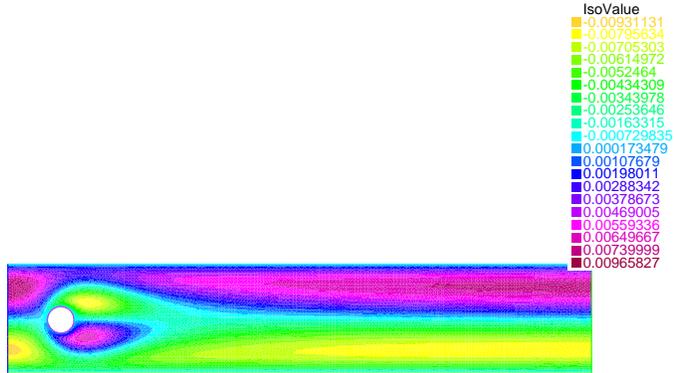


Figure 4.4: Streamline of flow around cylinder at the steady state with Reynolds number 20, mesh size $h_{max} = 1./60$, time step $\Delta t = 0.2h_{max}$. The problem was solved in non-dimensional variables, see Remark 3.14.

For $R_e = 20$, the flow turns into a stationary state. The following numbers are crucial in rigorous numerical analysis. They are:

- The drag coefficient

$$C_D = \frac{2F_D}{\rho U_{mean}^2 L}$$

- The lift coefficient

$$C_L = \frac{2F_L}{\rho U_{mean}^2 L}$$

- The pressure drop

$$\Delta p = p(A) - p(B) \quad \text{with} \quad A = (0.15, 0.2) \quad B = (0.25, 0.2)$$

For engineers, CFD is mostly applied to obtain the drag and lift coefficients for their further analysis. Usually the drag and lift forces are obtained from physical experiments first, then they can be calculated using the formulas mentioned above.

Suppose F_D and F_L are the drag and the lift forces respectively, they can be defined on the surface of the cylinder Γ_5 as follows:

$$\begin{pmatrix} F_D \\ F_L \end{pmatrix} = \int_{\Gamma_5} \sigma \vec{n} ds$$

where σ is the stress tensor, namely, $\sigma = -pI + \rho\mu\nabla\vec{u}$. We assume \vec{n} and $\vec{\tau}$ are the normal and the tangential vectors of the surface Γ_5 , namely, $\vec{n} = \begin{pmatrix} n_x \\ n_y \end{pmatrix}$ and $\vec{\tau} = \begin{pmatrix} n_y \\ -n_x \end{pmatrix}$. In order to express those forces as the line integrals on the surface Γ_5 , the stress tensor can be written in the following way:

$$\sigma = -pI + \rho\mu(D\vec{u}_t/D\vec{n})\vec{n}$$

This leads to the following expressions for F_D and F_L respectively:

$$F_D = \int_{\Gamma_5} \left(\frac{\partial \vec{u}_\tau}{\partial \vec{n}} n_y - p n_x \right) ds$$

$$F_L = - \int_{\Gamma_5} \left(\frac{\partial \vec{u}_\tau}{\partial \vec{n}} n_x + p n_y \right) ds$$

Those are still not-so-obvious calculations. We show the detailed calculation of the drag force F_D in the following equations, and the calculation of the lift force can be done in a similar way. The first term in the integrand of F_D can be expressed as:

$$\frac{\partial \vec{u}_\tau}{\partial \vec{n}} n_y = \left(\frac{\partial u_1}{\partial x} n_x n_y + \frac{\partial u_1}{\partial y} n_y^2 - \frac{\partial u_2}{\partial x} n_x^2 - \frac{\partial u_2}{\partial y} n_x n_y \right) n_y$$

Since we have the zero Dirichlet boundary condition on Γ_5 , namely, $\vec{u}|_{\Gamma_5} = \vec{0}$, it follows $\frac{\partial \vec{u}}{\partial \vec{r}} = \vec{0}$ on Γ_5 , in particular, $\frac{\partial u_1}{\partial x} n_y - \frac{\partial u_1}{\partial y} n_x = 0$ and $\frac{\partial u_2}{\partial x} n_y - \frac{\partial u_2}{\partial y} n_x = 0$. Since $n_x^2 + n_y^2 = 1$ and $\nabla \cdot \vec{u} = 0$, we have

$$\frac{\partial \vec{u}_\tau}{\partial \vec{n}} n_y = \left(\frac{\partial u_1}{\partial y} - \frac{\partial u_2}{\partial x} \right) n_y = \frac{\partial u_1}{\partial y} n_y - \frac{\partial u_2}{\partial y} n_x = \frac{\partial u_1}{\partial x} n_x + \frac{\partial u_1}{\partial y} n_y$$

Finally, the drag force can be explicitly calculated as:

$$F_D = \int_{\Gamma_5} \left(\rho \mu \left(\frac{\partial u_1}{\partial x} n_x + \frac{\partial u_1}{\partial y} n_y \right) - p n_x \right) ds$$

Similar analysis can be done for the lift force.

Convergence tests were performed on two mesh sizes: $h_{max} = 1/30$, and $h_{max} = 1/60$. The drag coefficient and the lift coefficient are plotted versus time in Figure (4.5) and (4.6). The reference data can be found in [34].

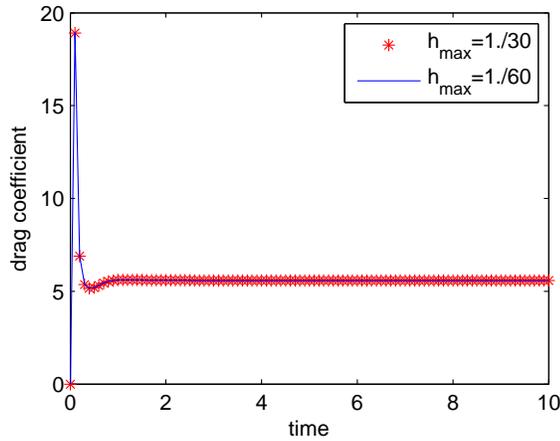


Figure 4.5: Evolution of the drag coefficient up to $t = 10$ with Reynolds number 20 under grid convergence tests. The problem was solved in non-dimensional variables, see Remark 3.14.

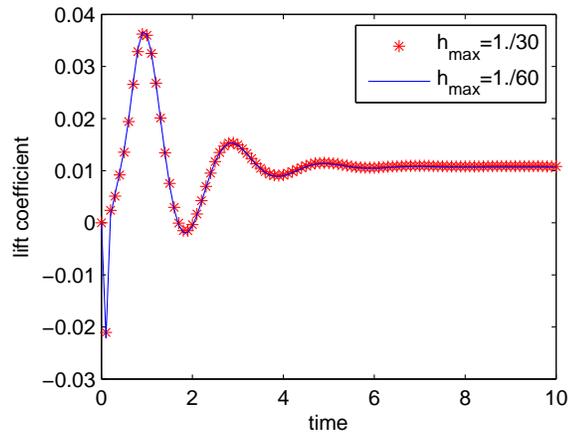


Figure 4.6: Evolution of the lift coefficient up to $t = 10$ with Reynolds number 20 under grid convergence tests. The problem was solved in non-dimensional variables, see Remark 3.14.

The pressure drop is $\Delta p = 0.1176$ under both mesh sizes. The drag coefficient, the lift coefficient, and the pressure drop agree well with the reference data, see [34].

We can increase the Reynolds number to $Re = 100$ by setting $U_{max} = 1.5$. We consider an unsteady flow instead in this case. Oscillations for the time revolutions of the drag and the lift can be observed, and this is the famous phenomenon named Vortex Induced Motion (VIM).

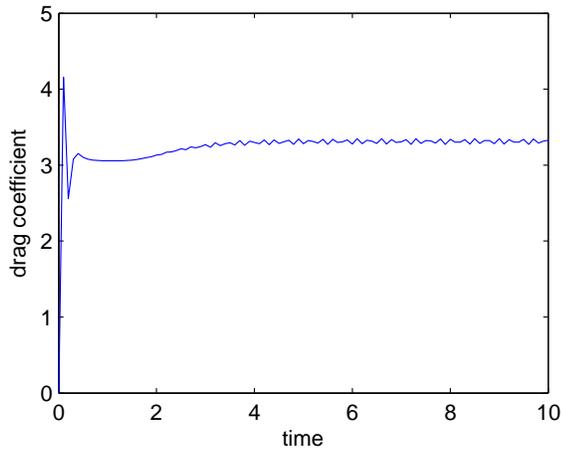


Figure 4.7: Evolution of the lift coefficient up to $t = 10$ with Reynolds number 100, mesh size $h_{max} = 1./60$, time step $\Delta t = 0.2h_{max}$. The problem was solved in non-dimensional variables, see Remark 3.14.

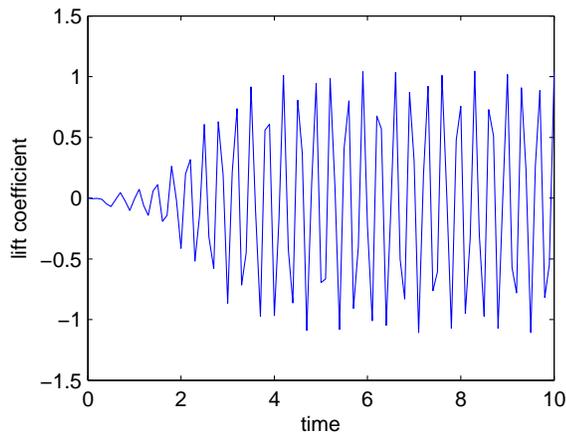


Figure 4.8: Evolution of the drag coefficient up to $t = 10$ with Reynolds number 100, mesh size $h_{max} = 1./60$, time step $\Delta t = 0.2h_{max}$. The problem was solved in non-dimensional variables, see Remark 3.14.

The corresponding pressure drop is $\Delta p = 2.5065$.

4.3 Driven cavity

Another classic benchmark problem for the one-phase incompressible fluid flow is the driven cavity problem. It models a plane of incompressible fluid flow in a lid-driven cavity. The computational domain is a unit square.

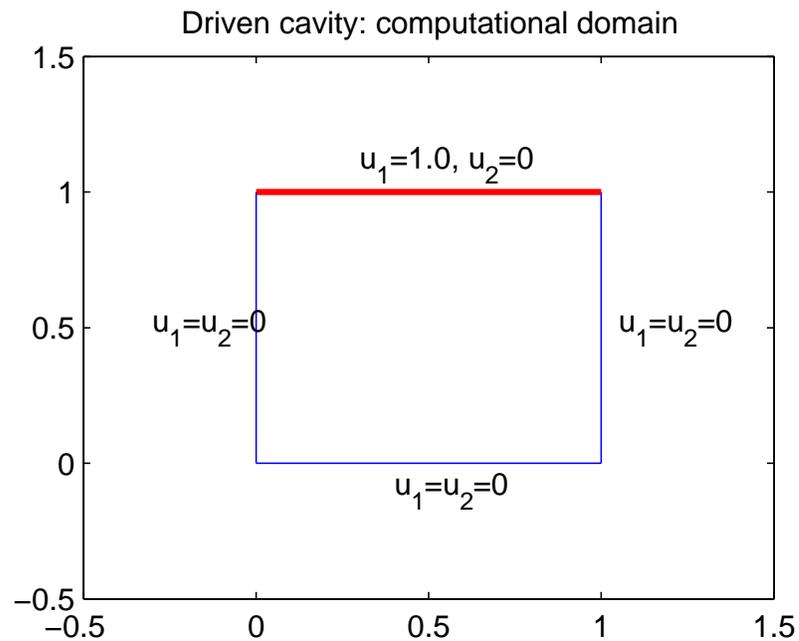


Figure 4.9: Driven cavity configuration. The problem was solved in non-dimensional variables, see Remark 3.14.

The statement of the problem is presented in Figure (4.9). The upper side of the cavity moves at a unit speed, and other sides are fixed. These settings lead to the following boundary conditions: zero Dirichlet boundary conditions are imposed on the left, the right and the bottom of the cavity; $u_1 = 1$, and $u_2 = 0$ are imposed on the top of cavity. There is a discontinuity on the top two corners of the cavity. We assumed those two points belong to the leaky cavity in all the simulations; however, one can also assume they are in the fixed vertical walls as an alternative approach. In other words, the Dirichlet boundary

conditions are prescribed everywhere on the boundaries for the velocity field. Since the pressure field is only known up to a constant as mentioned in previous chapters, a reference value $p = 0$ is imposed on the left bottom corner of the computational domain.

Firstly, one-phase fluid flow with a density $\rho = 1.0$ and a viscosity $\mu = 0.01$ was simulated with the mesh size $h = 1./40$ and the time step $\Delta t = 1./4h$. Simulations were ran up to a relative steady state, where the speed difference between two time steps was less than $1e - 03$. The three-step method, the two-step method, and the Van Kan scheme were implemented. The discontinuity of the velocity field on the upper two corners of the domain introduces a singularity in the pressure field precisely on those corners, see Figure (4.10) to (4.12).

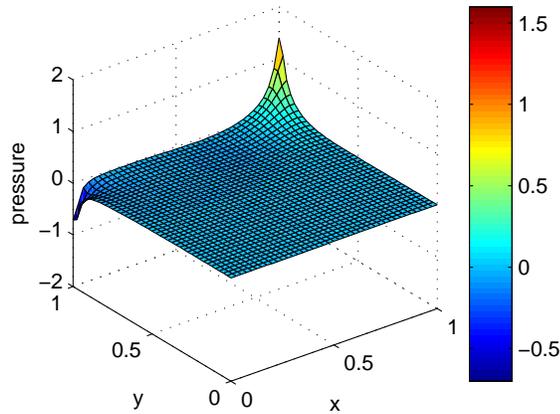


Figure 4.10: Pressure at the steady state from three-step method with mesh size $h = 1./40$ and time step $\Delta t = 1./4h$ using $\rho = 1.0$, $\mu = 0.01$. The problem was solved in non-dimensional variables, see Remark 3.14.

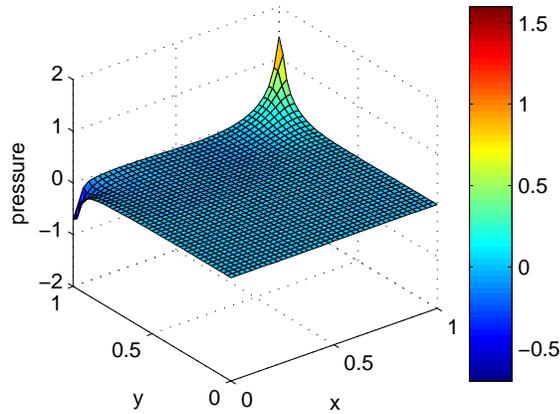


Figure 4.11: Pressure at the steady state from two-step method with mesh size $h = 1./40$ and time step $\Delta t = 1./4h$ using $\rho = 1.0$, $\mu = 0.01$. The problem was solved in non-dimensional variables, see Remark 3.14.

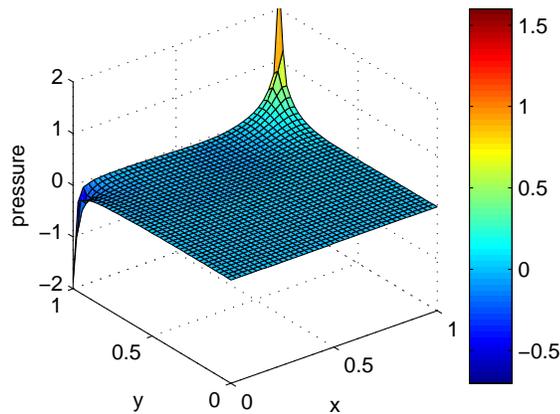


Figure 4.12: Pressure at the steady state from Van Kan method with mesh size $h = 1./40$ and time step $\Delta t = 1./4h$ using $\rho = 1.0$, $\mu = 0.01$. The problem was solved in non-dimensional variables, see Remark 3.14.

Figure (4.13) shows the velocity profile at the horizontal and the vertical center line respectively. The streamlines are reported in Figure (4.14) to (4.16).

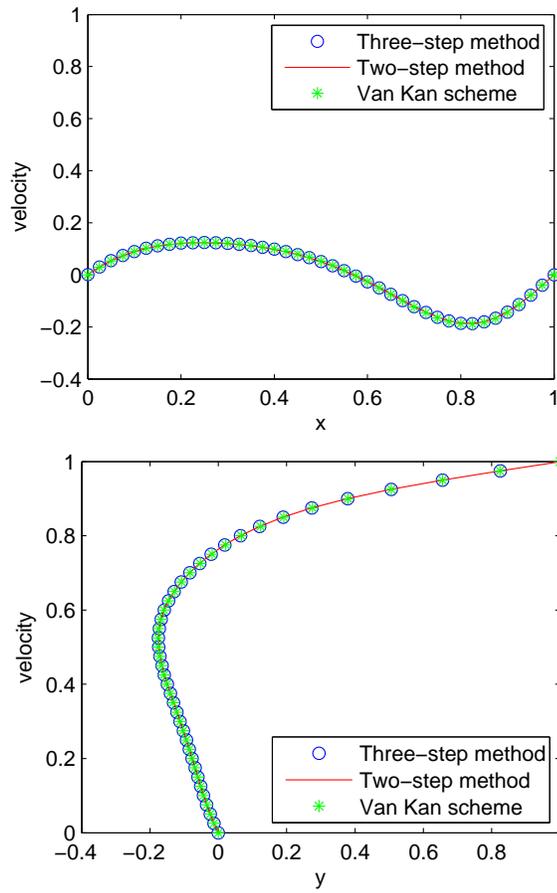


Figure 4.13: Velocity profiles at the horizontal and vertical center lines at the steady state with mesh size $h = 1./40$ and time step $\Delta t = 1./4h$ using $\rho = 1.0$, $\mu = 0.01$. The problem was solved in non-dimensional variables, see Remark 3.14.

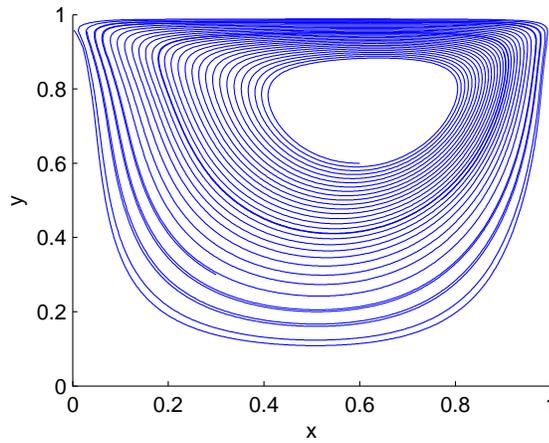


Figure 4.14: Streamline at the steady state from three-step method with mesh size $h = 1./40$ and time step $\Delta t = 1./4h$ using $\rho = 1.0$, $\mu = 0.01$. The problem was solved in non-dimensional variables, see Remark 3.14.

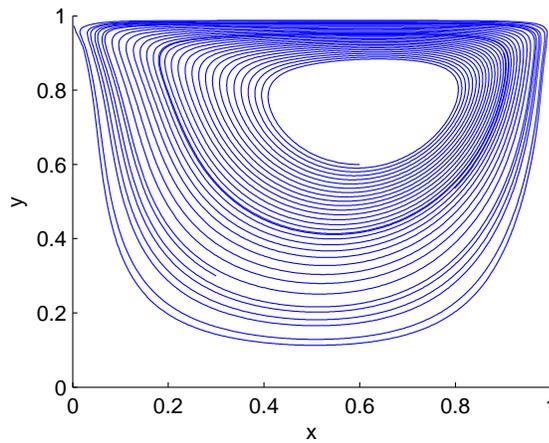


Figure 4.15: Streamline at the steady state from two-step method with mesh size $h = 1./40$ and time step $\Delta t = 1./4h$ using $\rho = 1.0$, $\mu = 0.01$. The problem was solved in non-dimensional variables, see Remark 3.14.

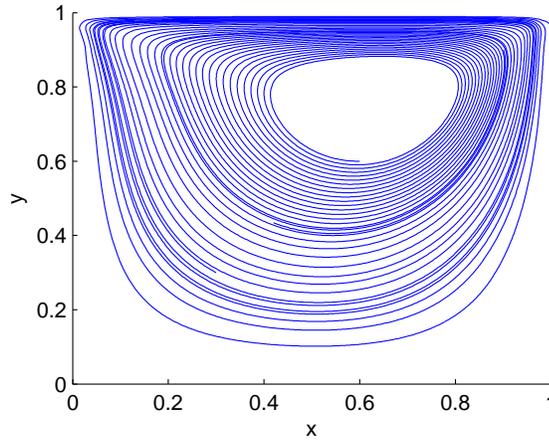


Figure 4.16: Streamline at the steady state from Van Kan scheme with mesh size $h = 1./40$ and time step $\Delta t = 1./4h$ using $\rho = 1.0$, $\mu = 0.01$. The problem was solved in non-dimensional variables, see Remark 3.14.

Next, we want to simulate a two-phase flow problem with different densities. The first fluid flows into the cavity with a density $\rho_1 = 0.1$. The density of the second flow is $\rho_2 = 1.0$. The second fluid initially stays in a circle centered on $(0.5, 0.5)$ with radius 0.25. The viscosities are the same for both fluid flows, and they are set to be $\mu_1 = \mu_2 = 0.01$. Our goal of this benchmark is to test the Volume Fraction method mentioned in Chapter 3. Mesh sizes were the same as those used in the one-phase driven cavity test, and all three methods for fluid flows were applied. Results are only computed up to $t = 5$ for illustration. The pressure field is presented in Figure (4.17) to (4.19) for all schemes.

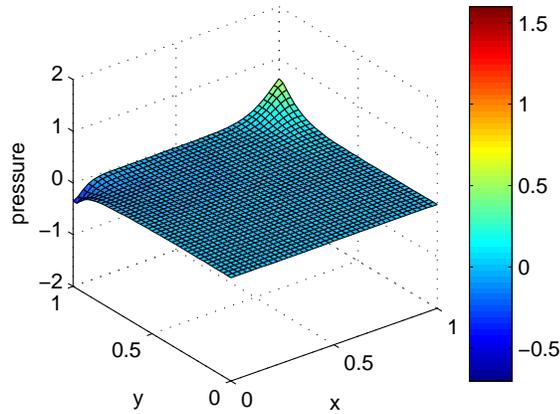


Figure 4.17: Pressure at $t = 5$ from three-step method with mesh size $h = 1./40$ and time step $\Delta t = 1./4h$ using $\rho_1 = 0.1, \rho_2 = 1.0, \mu_1 = \mu_2 = 0.01$. The problem was solved in non-dimensional variables, see Remark 3.14.

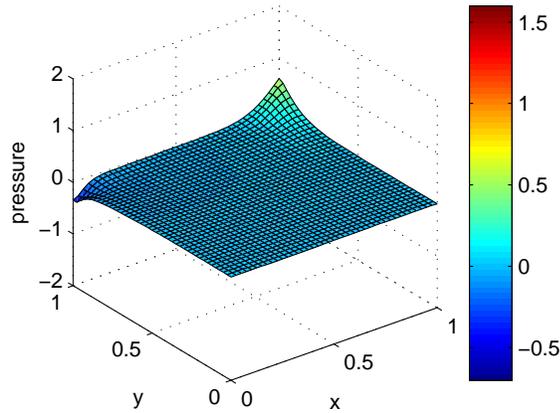


Figure 4.18: Pressure at $t = 5$ from two-step method with mesh size $h = 1./40$ and time step $\Delta t = 1./4h$ using $\rho_1 = 0.1, \rho_2 = 1.0, \mu_1 = \mu_2 = 0.01$. The problem was solved in non-dimensional variables, see Remark 3.14.

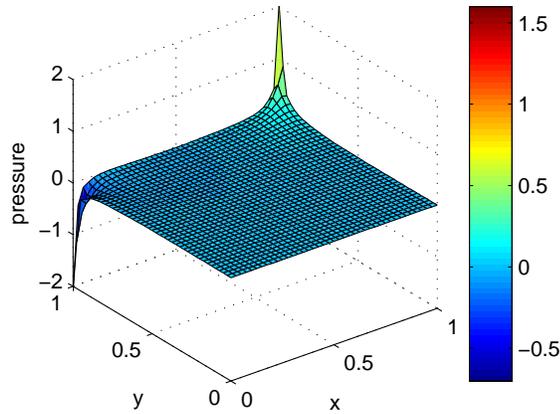


Figure 4.19: Pressure at $t = 5$ from Van Kan method with mesh size $h = 1./40$ and time step $\Delta t = 1./4h$ using $\rho_1 = 0.1, \rho_2 = 1.0, \mu_1 = \mu_2 = 0.01$. The problem was solved in non-dimensional variables, see Remark 3.14.

We can also observe the pressure singularities at the upper corners of our computational domain. The three-step method and the two-step method give a similar velocity profile, even though they solve the convection term in a different way. The Van Kan scheme give a flatter velocity at the horizontal center line. The velocity profile and the streamlines are reported in Figure (4.20) to (4.23).

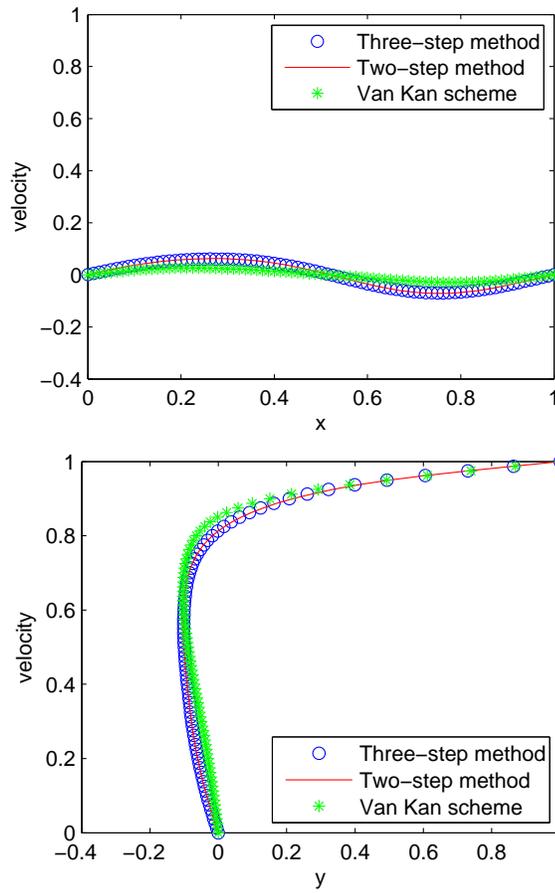


Figure 4.20: Velocity profiles at the horizontal and vertical center lines at $t = 5$ with mesh size $h = 1./40$ and time step $\Delta t = 1./4h$ using $\rho_1 = 0.1, \rho_2 = 1.0, \mu_1 = \mu_2 = 0.01$. The problem was solved in non-dimensional variables, see Remark 3.14.

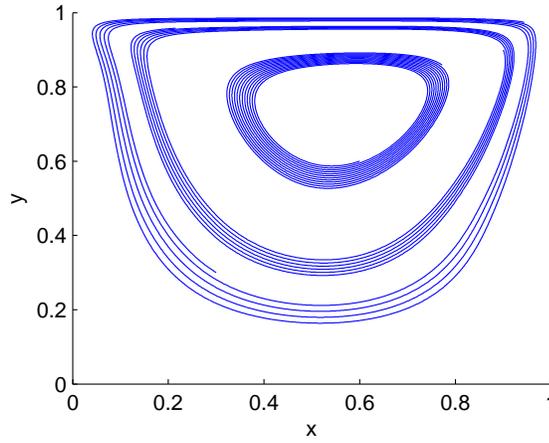


Figure 4.21: Streamline at $t = 5$ from three-step method with mesh size $h = 1./40$ and time step $\Delta t = 1./4h$ using $\rho_1 = 0.1, \rho_2 = 1.0, \mu_1 = \mu_2 = 0.01$. The problem was solved in non-dimensional variables, see Remark 3.14.

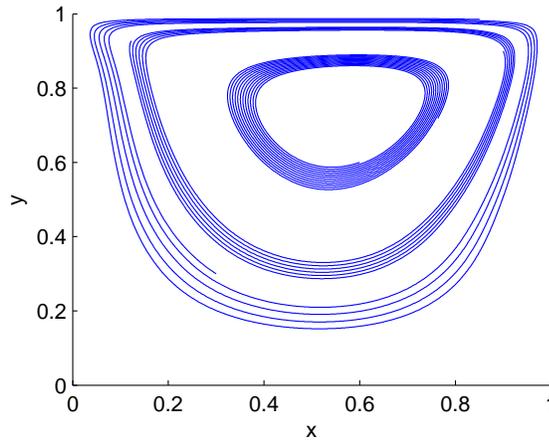


Figure 4.22: Streamline at $t = 5$ from two-step method with mesh size $h = 1./40$ and time step $\Delta t = 1./4h$ using $\rho_1 = 0.1, \rho_2 = 1.0, \mu_1 = \mu_2 = 0.01$. The problem was solved in non-dimensional variables, see Remark 3.14.

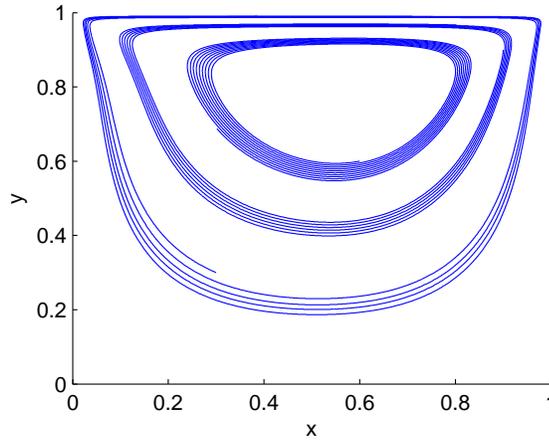


Figure 4.23: Streamline at $t = 5$ from Van Kan scheme with mesh size $h = 1./40$ and time step $\Delta t = 1./4h$ using $\rho_1 = 0.1, \rho_2 = 1.0, \mu_1 = \mu_2 = 0.01$. The problem was solved in non-dimensional variables, see Remark 3.14.

Simulating a two-phase flow problem with different viscosities is a much more challenging work, even though the fluid flows have same densities. We consider a fluid flow with lower viscosity flows into the driven cavity in this case. No $\nabla \vec{u}^T$ term presents in the weak formulations of the previous two cases, because the flows have same viscosities. This term cannot be avoided in this scenario. We can also apply the Volume Fraction method to evaluate the different viscosities. Numerical results are presented in Figure (4.24) to Figure (4.30) for the pressure field, the velocity profile at the horizontal and the vertical center line, and the streamlines.

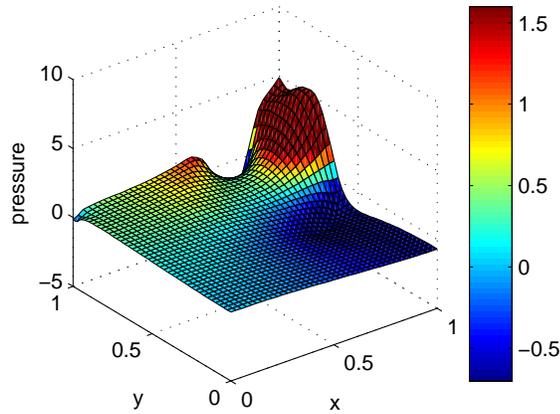


Figure 4.24: Pressure at $t = 5$ from three-step method with mesh size $h = 1./40$ and time step $\Delta t = 1./4h$ using $\rho_1 = \rho_2 = 1.0$, $\mu_1 = 0.01$, $\mu_2 = 10.0$. The problem was solved in non-dimensional variables, see Remark 3.14.

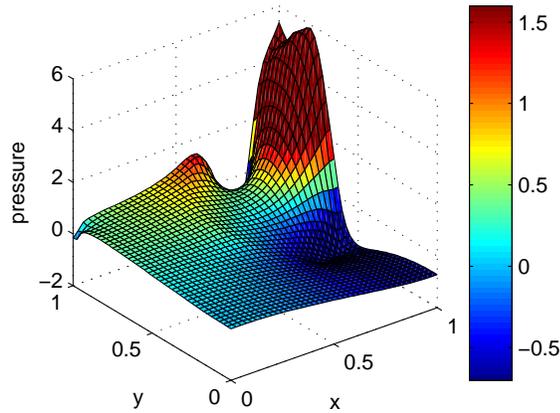


Figure 4.25: Pressure at $t = 5$ from two-step method with mesh size $h = 1./40$ and time step $\Delta t = 1./4h$ using $\rho_1 = \rho_2 = 1.0$, $\mu_1 = 0.01$, $\mu_2 = 10.0$. The problem was solved in non-dimensional variables, see Remark 3.14.

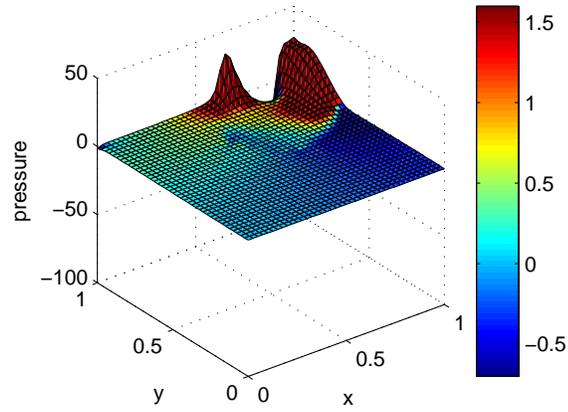


Figure 4.26: Pressure at $t = 5$ from Van Kan method with mesh size $h = 1./40$ and time step $\Delta t = 1./4h$ using $\rho_1 = \rho_2 = 1.0$, $\mu_1 = 0.01$, $\mu_2 = 10.0$. The problem was solved in non-dimensional variables, see Remark 3.14.

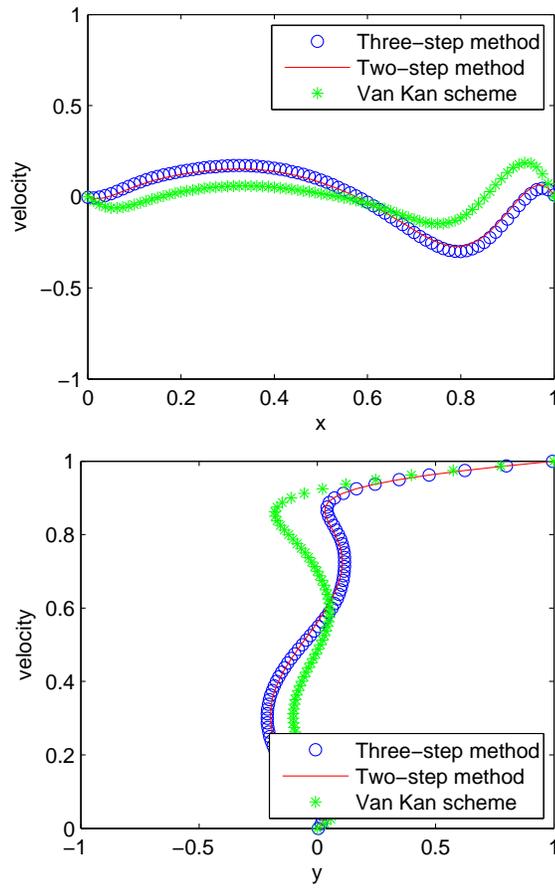


Figure 4.27: Velocity profiles at the horizontal and vertical center lines at $t = 5$ with mesh size $h = 1./40$ and time step $\Delta t = 1./4h$ using $\rho_1 = \rho_2 = 1.0$, $\mu_1 = 0.01$, $\mu_2 = 10.0$. The problem was solved in non-dimensional variables, see Remark 3.14.

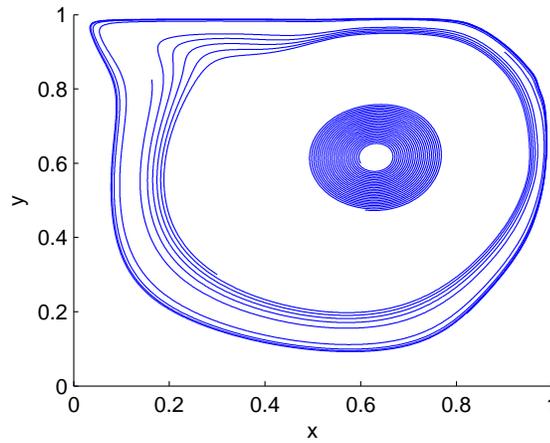


Figure 4.28: Streamline at $t = 5$ from three-step method with mesh size $h = 1./40$ and time step $\Delta t = 1./4h$ using $\rho_1 = \rho_2 = 1.0$, $\mu_1 = 0.01$, $\mu_2 = 10.0$. The problem was solved in non-dimensional variables, see Remark 3.14.

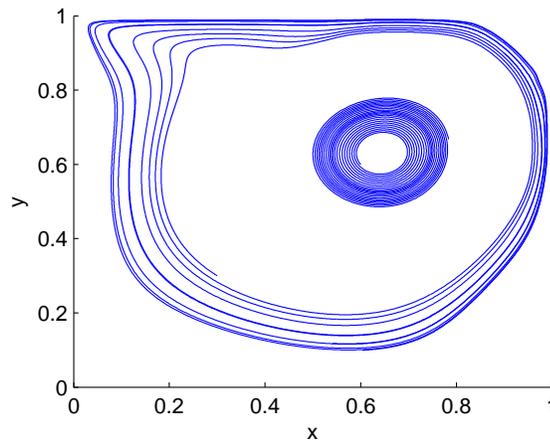


Figure 4.29: Streamline at $t = 5$ from two-step method with mesh size $h = 1./40$ and time step $\Delta t = 1./4h$ using $\rho_1 = \rho_2 = 1.0$, $\mu_1 = 0.01$, $\mu_2 = 10.0$. The problem was solved in non-dimensional variables, see Remark 3.14.

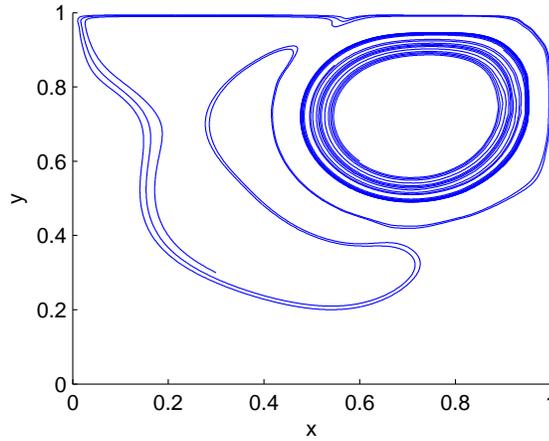


Figure 4.30: Streamline at $t = 5$ from Van Kan scheme with mesh size $h = 1./40$ and time step $\Delta t = 1./4h$ using $\rho_1 = \rho_2 = 1.0$, $\mu_1 = 0.01$, $\mu_2 = 10.0$. The problem was solved in non-dimensional variables, see Remark 3.14.

Results coincide for the three-step method and the two-step method since there is no pressure associated with the convection and the diffusion process. All three methods give more singularities of the pressure profile, especially on the upper right corner in the cavity due to the large ratio of viscosities ($\mu_2/\mu_1 = 1000$).

4.4 Static bubble

This test case models a circular bubble in a perfect equilibrium state. According to the Laplace-Young law, the pressure differential is proportional to the surface tension coefficient, and it is inversely proportional to the radius of the bubble, namely, $p_{in} - p_{out} = \sigma/R$. Since everything is stationary, the correct solution for the velocity field should be zero. However, some spurious velocity currents will be generated due to certain imbalances in the numerical schemes. Lafaurie et al. [67] proposed the amplitude of this spurious velocity current is proportional to σ/μ , which is equivalent to having an approximately constant

capillary number $Ca = U\mu/\sigma$.

The computational domain is a unit square. A bubble with radius $R = 0.25$ is centered at $(0.5, 0.5)$ initially. The density and viscosity ratios are equal to 1, namely $\rho_1 = \rho_2 = 10^4$, and $\mu_1 = \mu_2 = 1$. The no-slip Dirichlet homogeneous boundary conditions are imposed everywhere on the boundaries for the velocity field. The reference pressure $p_{out} = 0$ is set at the upper right corner of our computational domain. The gravity is neglected. The surface tension coefficient may be varied. We set this parameter to be 1, which gives the Laplace number $La = 2R\sigma\rho/\mu^2 = 5 \times 10^3$. The P2-P1 Finite Element was used for the discretization of the velocity and the pressure. The simulations were performed until $t = 3$ with a fixed time step $\Delta t = 0.01$. The three-step method, the two-step method, and the Van Kan scheme were implemented for the fluid flows. Only the linearized version of these schemes were applied, because the flows have same densities and viscosities.

The convergence results for the non-dimensional velocity in two different norms are listed in Table 4.3 and 4.4. The l^∞ norm is defined as $\max|\vec{u}\mu/\sigma|$, and the l^1 norm is defined as $1/N \sum_{i=1}^N |\vec{u}_i\mu/\sigma|$, where N is the number of degrees of freedom for the velocity field. The two-step method and the three-step method give more accuracy in l^∞ norm. Again, the two-step method and the three-step method are essentially the same in evaluating the surface tension forces. The presence of the pressure term in the first step of the Van Kan scheme makes the splitting of the surface tension forces more complicated, thus it requires a more careful choice of the time step. However, all three methods give same results in l^1 norm.

Table 4.3: Errors in the l^∞ norm for the non-dimensional velocity

h	Three-step method	Two-step method	Van Kan scheme
1./20	$3.51 \cdot 10^{-2}$	$3.52 \cdot 10^{-2}$	$3.52 \cdot 10^{-2}$
1./40	$2.22 \cdot 10^{-2}$	$2.23 \cdot 10^{-2}$	$2.23 \cdot 10^{-2}$
1./80	$1.35 \cdot 10^{-2}$	$1.35 \cdot 10^{-2}$	$1.37 \cdot 10^{-2}$

Table 4.4: Errors in the l^1 norm for the non-dimensional velocity

h	Three-step method	Two-step method	Van Kan scheme
1./20	$1.5 \cdot 10^{-3}$	$1.5 \cdot 10^{-3}$	$1.5 \cdot 10^{-3}$
1./40	$5.0 \cdot 10^{-4}$	$5.0 \cdot 10^{-4}$	$5.0 \cdot 10^{-4}$
1./80	$1.0 \cdot 10^{-4}$	$1.0 \cdot 10^{-4}$	$1.0 \cdot 10^{-4}$

Figure (4.31) and Figure (4.32) show the loglog plot of the errors in l^∞ and l^1 norms.

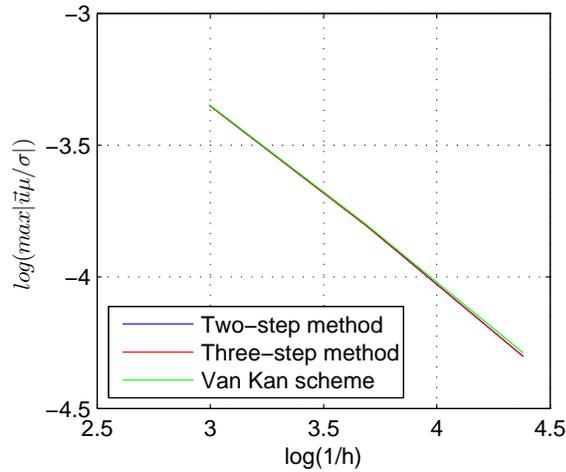


Figure 4.31: Comparison of three-step method, two-step method and Van Kan scheme for l^∞ error of dimensionless velocity at $t = 3$. The problem was solved in non-dimensional variables, see Remark 3.14.

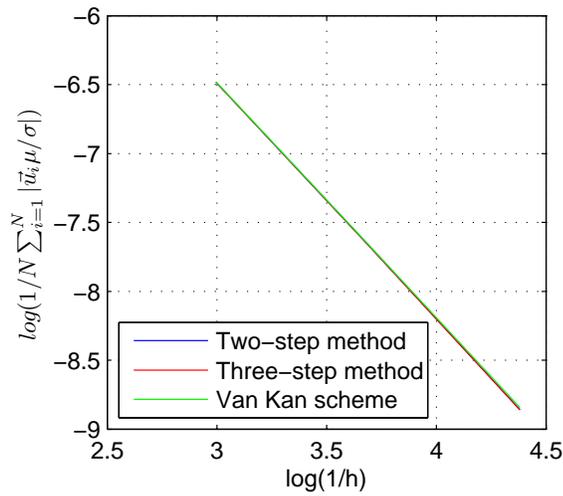


Figure 4.32: Comparison of three-step method, two-step method and Van Kan scheme for l^1 error of dimensionless velocity at $t = 3$. The problem was solved in non-dimensional variables, see Remark 3.14.

Agreement of the three-step method and the two-step method is further verified by the pressure behavior in the following figures. Figure (4.33) to (4.35) show the pressure cut-lines at $y = 0.5$ using three methods with different mesh sizes.

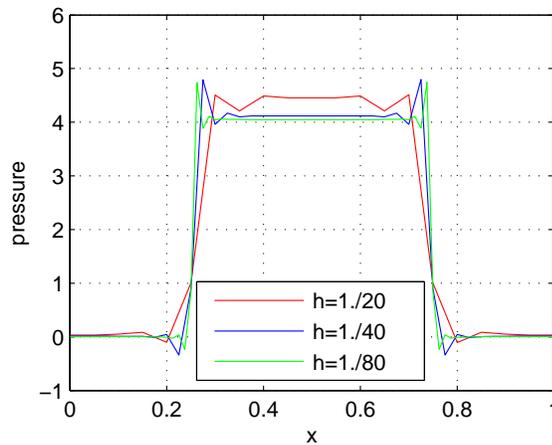


Figure 4.33: Pressure cut-line at $y = 0.5$ using three-step method under grid refinement at $t = 3$. The problem was solved in non-dimensional variables, see Remark 3.14.

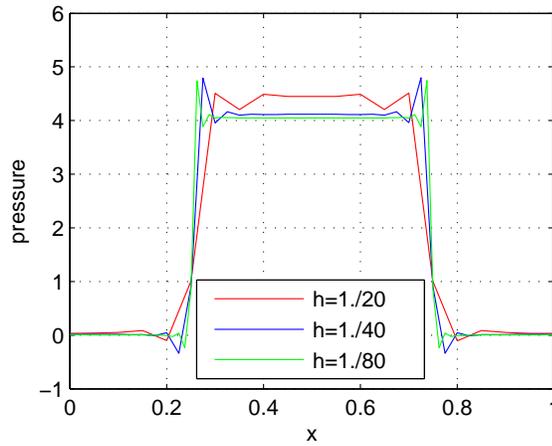


Figure 4.34: Pressure cut-line at $y = 0.5$ using two-step method under grid refinement at $t = 3$. The problem was solved in non-dimensional variables, see Remark 3.14.

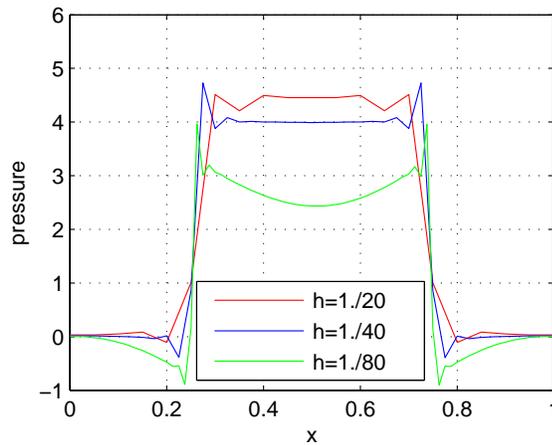


Figure 4.35: Pressure cut-line at $y = 0.5$ using Van Kan scheme under grid refinement at $t = 3$. The problem was solved in non-dimensional variables, see Remark 3.14.

Spikes near the edge of the interface are noticed, and this is because we use the continuous approximations for the pressure field in all methods. However, the Van Kan scheme is unstable with a finer mesh. We can overcome this problem by using a smaller time step. We will see this in the following oscillating bubble benchmark problem. Results from the two-step method and the three-step method justify the splitting of the surface tension

effect.

The agreement of the Laplace-Young law is further presented in the following absolute and relative errors of the pressure for the two-step method and the three-step method.

Table 4.5: Absolute errors $|p_{in} - p_{out} - \sigma/R|$ in the Laplace-Young law

h	Three-step method	Two-step method
1./20	$3.45 \cdot 10^{-1}$	$3.62 \cdot 10^{-1}$
1./40	$1.04 \cdot 10^{-1}$	$1.02 \cdot 10^{-1}$
1./80	$4.27 \cdot 10^{-2}$	$4.10 \cdot 10^{-2}$

Table 4.6: Relative errors $\frac{|p_{in} - p_{out} - \sigma/R|}{\sigma/R}$ in the Laplace-Young law

h	Three-step method	Two-step method
1./20	$9.13 \cdot 10^{-2}$	$9.08 \cdot 10^{-2}$
1./40	$2.61 \cdot 10^{-2}$	$2.53 \cdot 10^{-2}$
1./80	$1.07 \cdot 10^{-2}$	$1.03 \cdot 10^{-2}$

The pressure results might not be ideal, but Smolianski's presented similar results in his work, see [52]. In addition, Figure (4.33) to (4.35) show the surface tension forces are coupled correctly with the fluid flows.

4.5 Oscillating bubble

In this oscillating bubble test, the previous test configuration is made a little more difficult. Initially, the original circular bubble is perturbed to be an elliptic bubble by scaling the semi-axes a factor 1.25 in the x-direction and 0.8 in the y-direction. The elliptic bubble is also centered in the unit square. The bubble consists of the same fluid as in the surrounding

unit square. The fluid has a density 10^4 , a viscosity 1, and the surface tension coefficient is set to be 0.1. We simulated the behaviors of the elliptic bubble up to time=1000. At the final time, the bubble is expected to have reached an equilibrium state, where the bubble has a stable circular shape. The grid refinements were applied both in space and in the time scale. Although the surface tension effect was implemented using the semi-implicit approach, the theoretical capillary time step restriction $\sqrt{\rho h^3/\sigma}$ was used for the stability concerns. The three-step method, the two-step method, and the Van Kan scheme were implemented with the P2-P1 Finite Element pair for the velocity and the pressure field. Table 4.7 presents the time step sizes with different spatial refinement levels.

Table 4.7: mesh sizes of grid refinement level oscillating bubble

h	1./20	1./40	1./80
Δt	1.4	0.5	0.17

Figure (4.36) to (4.38) show results from the three-step method at six different times $t = 0, 90, 180, 260, 330,$ and 1000 using the mesh sizes mentioned in Table 4.7.

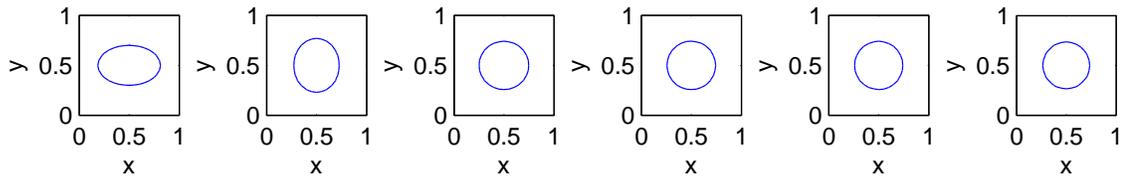


Figure 4.36: Oscillating bubble at $t = 0, 90, 180, 260, 330,$ and 1000 using three-step method with mesh size $h = 1./20$ and time step $\Delta t = 1.4$. The problem was solved in non-dimensional variables, see Remark 3.14.

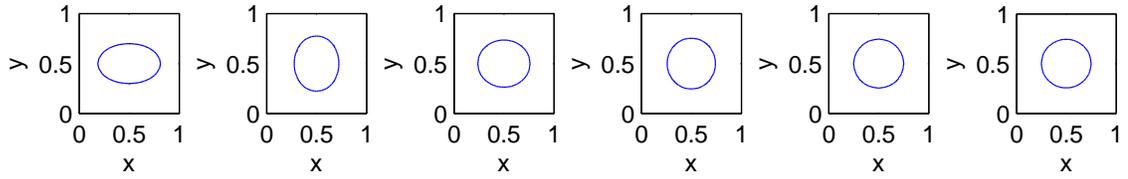


Figure 4.37: Oscillating bubble at $t = 0, 90, 180, 260, 330,$ and 1000 using three-step method with mesh size $h = 1./40$ and time step $\Delta t = 0.5$. The problem was solved in non-dimensional variables, see Remark 3.14.

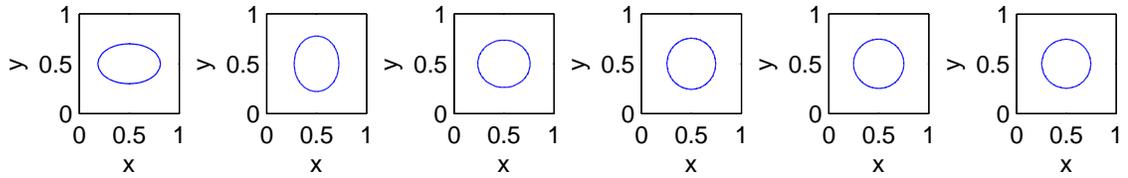


Figure 4.38: Oscillating bubble at $t = 0, 90, 180, 260, 330,$ and 1000 using three-step method with mesh size $h = 1./80$ and time step $\Delta t = 0.17$. The problem was solved in non-dimensional variables, see Remark 3.14.

Evolutions of the bubble using the two-step method and the Van Kan scheme are presented in Figure (4.39) to (4.44). All three methods give reasonable results under these grid refinements.

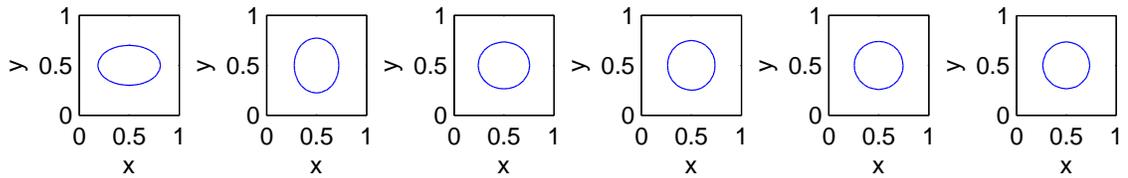


Figure 4.39: Oscillating bubble at $t = 0, 90, 180, 260, 330,$ and 1000 using two-step method with mesh size $h = 1./20$ and time step $\Delta t = 1.4$. The problem was solved in non-dimensional variables, see Remark 3.14.

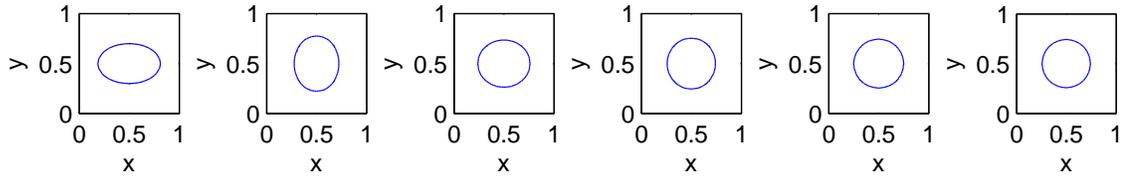


Figure 4.40: Oscillating bubble at $t = 0, 90, 180, 260, 330,$ and 1000 using two-step method with mesh size $h = 1./40$ and time step $\Delta t = 0.5$. The problem was solved in non-dimensional variables, see Remark 3.14.

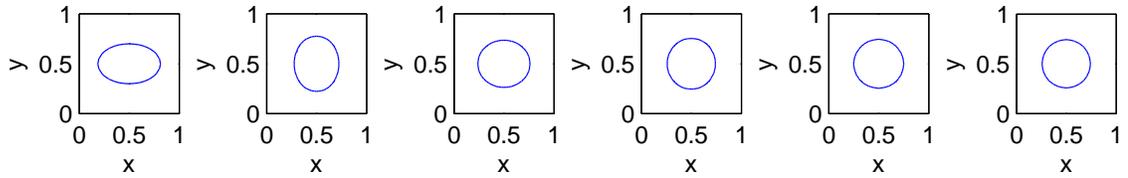


Figure 4.41: Oscillating bubble at $t = 0, 90, 180, 260, 330,$ and 1000 using two-step method with mesh size $h = 1./80$ and time step $\Delta t = 0.17$. The problem was solved in non-dimensional variables, see Remark 3.14.

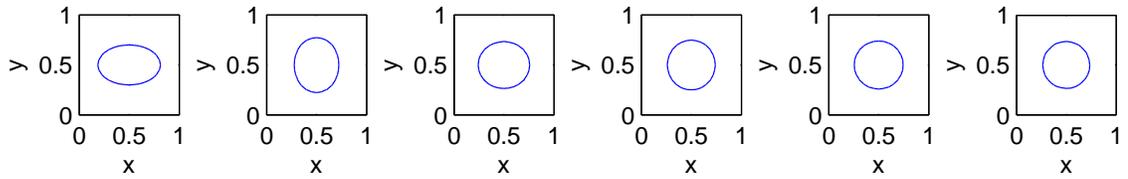


Figure 4.42: Oscillating bubble at $t = 0, 90, 180, 260, 330,$ and 1000 using Van Kan scheme with mesh size $h = 1./20$ and time step $\Delta t = 1.4$. The problem was solved in non-dimensional variables, see Remark 3.14.

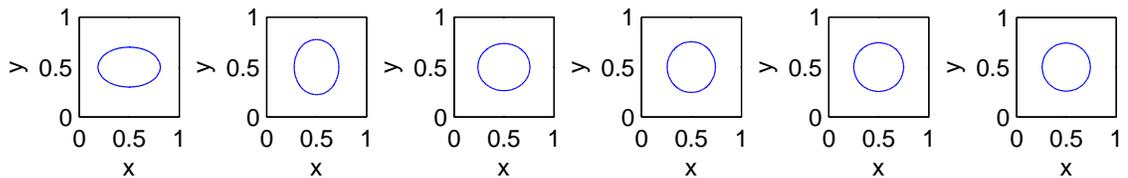


Figure 4.43: Oscillating bubble at $t = 0, 90, 180, 260, 330,$ and 1000 using Van Kan scheme with mesh size $h = 1./40$ and time step $\Delta t = 0.5$. The problem was solved in non-dimensional variables, see Remark 3.14.

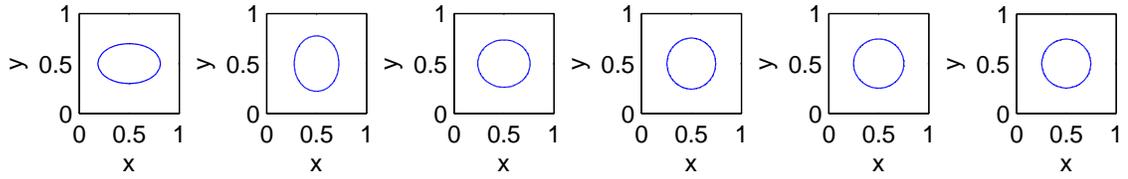


Figure 4.44: Oscillating bubble at $t = 0, 90, 180, 260, 330,$ and 1000 using Van Kan scheme with mesh size $h = 1./80$ and time step $\Delta t = 0.17$. The problem was solved in non-dimensional variables, see Remark 3.14.

4.6 Rising bubble

Our final test is the simulation of a single bubble. The bubble rises through the surrounding fluid due to the buoyancy effect. Initially, there is a circular bubble centered at $(0.5, 0.5)$ with radius $R_0 = 0.25$ in the container. The container is a rectangle with width 1 and height 2, see Figure (4.45).

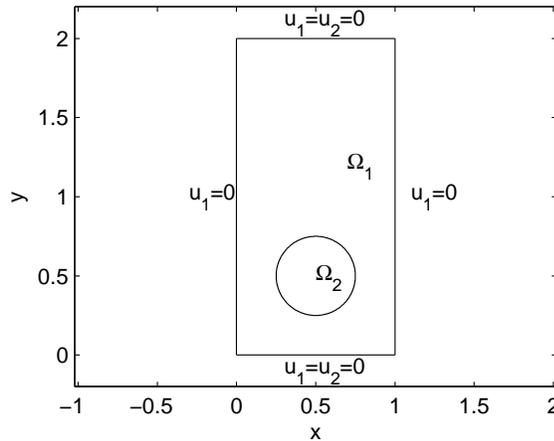


Figure 4.45: Initial configuration of rising bubble test. The problem was solved in non-dimensional variables, see Remark 3.14.

The bubble has a density $\rho_2 = 100$ and a viscosity $\mu_2 = 1$. The surrounding fluid has a density of $\rho_1 = 1000$, and viscosity $\mu_1 = 10$. The no-slip boundary conditions, namely

$\vec{u} = \vec{0}$, are imposed on the top and the bottom of the boundaries. The vertical walls are the symmetry boundaries, where the free-flip boundary conditions are imposed, namely, $\vec{u} \cdot \vec{n} = 0$, $\vec{\tau} \cdot (\nabla \vec{u} + \nabla \vec{u}^T) \cdot \vec{n} = 0$ (here $\vec{\tau}$ is the tangential vector). The gravitational constant is set to be $g = -0.98$. The surface tension coefficient is set to be $\sigma = 24.5$. The problem is characterized by the following Reynolds (R_e) and Eötvös (E_o) numbers:

$$R_e = \frac{\rho_1 \sqrt{g} (2R_0)^{3/2}}{\mu_1}, \quad E_o = \frac{4\rho_1 g R_0^2}{\sigma}$$

The density ratio ρ_1/ρ_2 and the viscosity ratio μ_1/μ_2 are two additional dimensionless numbers to fully characterize this test problem. All physical parameters and the dimensionless numbers defining the test case are summarized in Table (4.8). The problem was solved in non-dimensional variables.

Table 4.8: Physical parameters and dimensionless numbers of rising bubble test

ρ_1	ρ_2	μ_1	μ_2	g	σ	ρ_1/ρ_2	μ_1/μ_2	Re	E_o
1000	100	10	1	-0.98	24.5	10	10	35	10

The time evolutions of the bubble were tracked up to $t = 3.0$. The picture norms and the numerical quantities are measured. With all these physical parameters settings, the bubble shape can be predicted as a function of the Reynolds and Eötvös numbers using the experimental results from Clift, Grace, and Weber [63]. Although the experimental results were obtained in a three-dimensional regime, they provide a guideline for the two-dimensional case. According to their studies, the bubble will end up with a ellipsoidal shape, where the surface tension forces are strong enough to hold the bubble so that it will not break up. The picture norm is a first vital way to verify if our numerical schemes are reliable or not. However, the benchmark quantities are even more important in a rigorous

analysis. The following quantities are introduced as the measurements of accuracy for our numerical simulations.

Center of Mass:

The first quantity is the center of mass. Evolutions of the bubble can be tracked using the positions of various points. It is common to track the centroid or center of mass, see [25], [35], [33], [42]. The center of mass is defined as follows:

$$y_c = \frac{\int_{\Omega_2} y d\Omega}{\int_{\Omega_2} 1 d\Omega} \quad (4.3)$$

Here, Ω_2 denotes the region that the bubble occupies. We only track the vertical component due to the buoyancy force. The denominator denotes the area that the bubble occupies. This quantity can be easily extended into a three-dimensional parameter by carrying out the integrations in volume. Other points can also be used as the benchmark references, for example, Chen et al. suggested to use the absolute top or bottom of a bubble as a reference value, see [25].

Rise velocity:

The second quantity is the rise velocity or the mean velocity. This quantity is a particularly interesting one, since it measures the performance of the interface-tracking algorithm and the quality of the overall solution. The mean bubble velocity is defined as follows:

$$u_{2c} = \frac{\int_{\Omega_2} u_2 d\Omega}{\int_{\Omega_2} 1 d\Omega} \quad (4.4)$$

Here, Ω_2 also denotes the region that the bubble occupies, and u_2 is the vertical component of the velocity field. The vertical component of the velocity field at the center of the bubble $u_{2c}(\vec{x}_c)$ could be an alternative to evaluate the mean velocity. In addition, the terminal

velocities at the end of the time span could be meaningful information, see [51].

Circularity:

The last quantity is the circularity. The "degree of circularity" in two dimensions, introduced by Wadell [62], can be defined as follows:

$$c = \frac{P_a}{P_b} = \frac{\text{perimeter of area-equivalent circle}}{\text{perimeter of bubble}} = \frac{\pi d_a}{P_b} \quad (4.5)$$

Here, P_b denotes the perimeter or the circumference of a circle with diameter d_a which has an area equal to that of a bubble with a perimeter P_b . For a perfectly circular bubble, the circularity will be equal to unity and decrease as the bubble is deformed. We need some numerical approximations for the interfacial locations to calculate the perimeter of the bubble.

Initial benchmark studies attributed to the work of three research groups:

Group and Affiliation	Code/Method
1. University of Dortmund, Institute of Math. by S. Turek, D. Kuzmin, S. Hysing	TP2D FEM-Level set
2. EPFL Lausanne, Institute of Analysis and Scientific computing by E. Burman, N. Parolini	FREELIFE FEM-Level set
3. University of Magdeburg, Institute of Analysis and Numerical Math. by L. Tobiska, S. Ganesan	MooNMD FEM-ALE

The computational strategies and the convergence results obtained by these solvers are reviewed by Hysing in [20]. We only refer to their results as the reference data. We implemented the three-step method, the two-step method, and the Van Kan scheme as the

time integration techniques. The spatial discretizations were based on the P2-P1 Finite Element approach. We did not include any artificial stabilizations for the convective term in the momentum equations. The new Volume Fraction method was applied in order to evaluate the discontinuous coefficients. The linearized and non-linearized time integration techniques for the corresponding subproblems were investigated. Structured meshes with size $h = 1./20$, $h = 1./40$, and $h = 1./80$ were used for the convergence studies. The time step was taken with $\Delta t = 1./16h$. The mesh statistics are listed in Table (4.9). NOE refers to the number of elements. DOFU refers to the number of degrees of freedom for the velocity, and DFFP denotes the number of degrees of freedom for the pressure.

Table 4.9: Mesh statistics

mesh size	NOE	DOFU	DFFP
1./20	1600	3321	861
1./40	6400	13041	3321
1./80	25600	51681	13041

The following results show the grid convergence tests from all schemes for the bubble shapes at time $t = 3.0$. At the final time, all methods confirm the reference data with mesh size $h = 1./40$, and $h = 1./80$. Mass is not conserved in the coarser mesh $h = 1./20$.

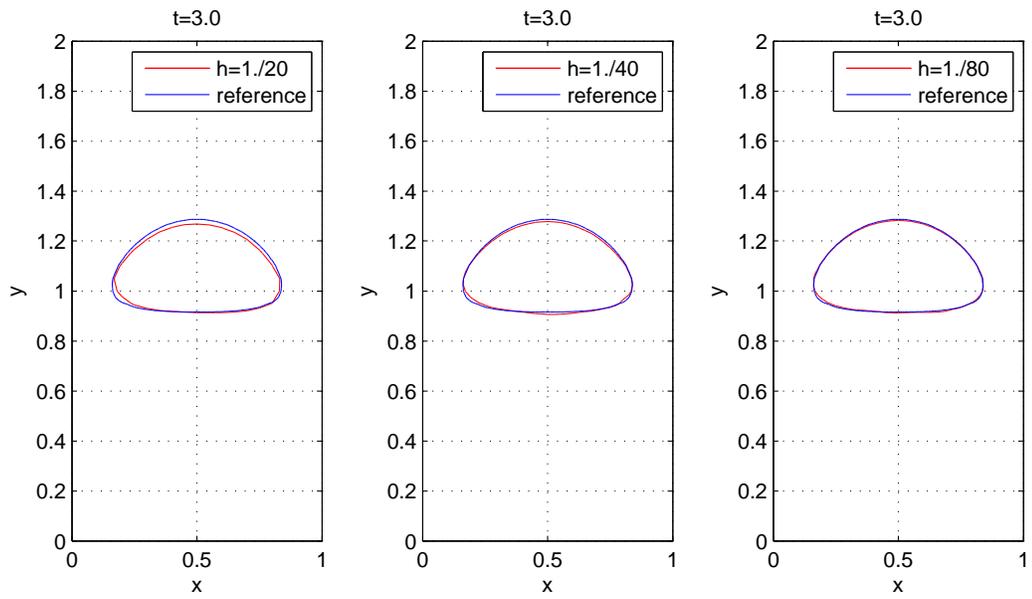


Figure 4.46: Bubble shapes under convergence tests from linearized three-step method with $\Delta t = 1./16h$ at $t = 3.0$ (reference solutions are obtained from TP2D code). The problem was solved in non-dimensional variables, see Remark 3.14.

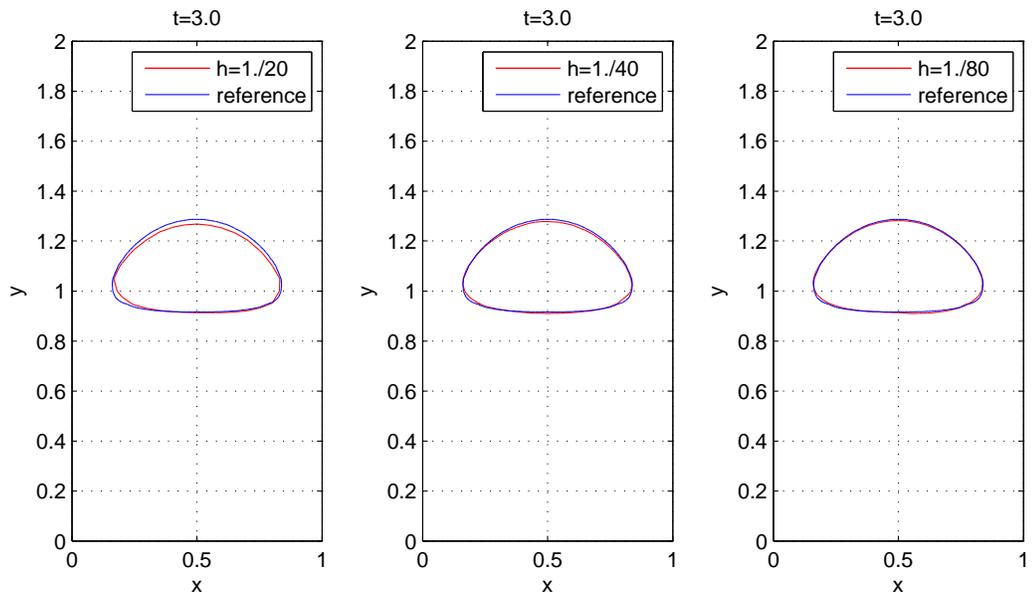


Figure 4.47: Bubble shapes under convergence tests from linearized two-step method with $\Delta t = 1./16h$ at $t = 3.0$ (reference solutions are obtained from TP2D code). The problem was solved in non-dimensional variables, see Remark 3.14.

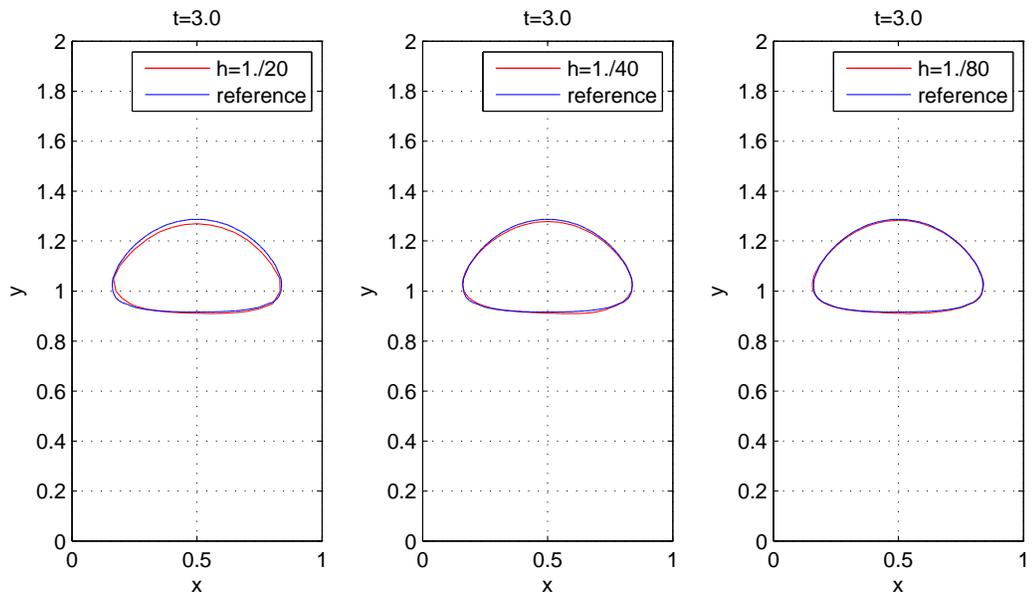


Figure 4.48: Bubble shapes under convergence tests from linearized Van Kan method with $\Delta t = 1./16h$ at $t = 3.0$ (reference solutions are obtained from TP2D code). The problem was solved in non-dimensional variables, see Remark 3.14.

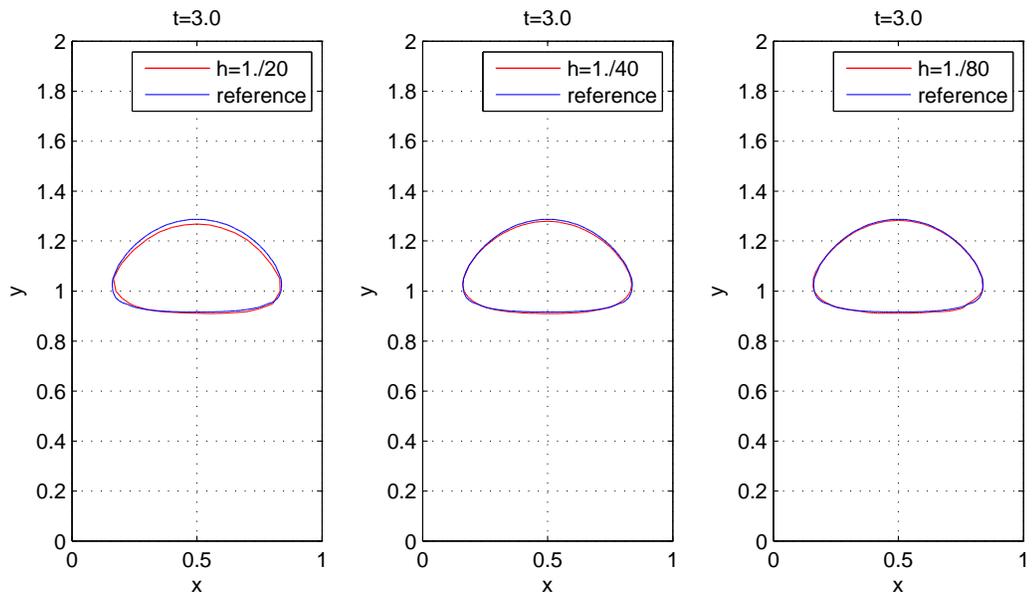


Figure 4.49: Bubble shapes under convergence tests from non-linearized three-step method with $\Delta t = 1./16h$ at $t = 3.0$ (reference solutions are obtained from TP2D code). The problem was solved in non-dimensional variables, see Remark 3.14.

The time evolutions of the interface shapes at different times for all schemes are listed in Figure (4.50) to (4.57). All numerical schemes are stable in these picture norms.

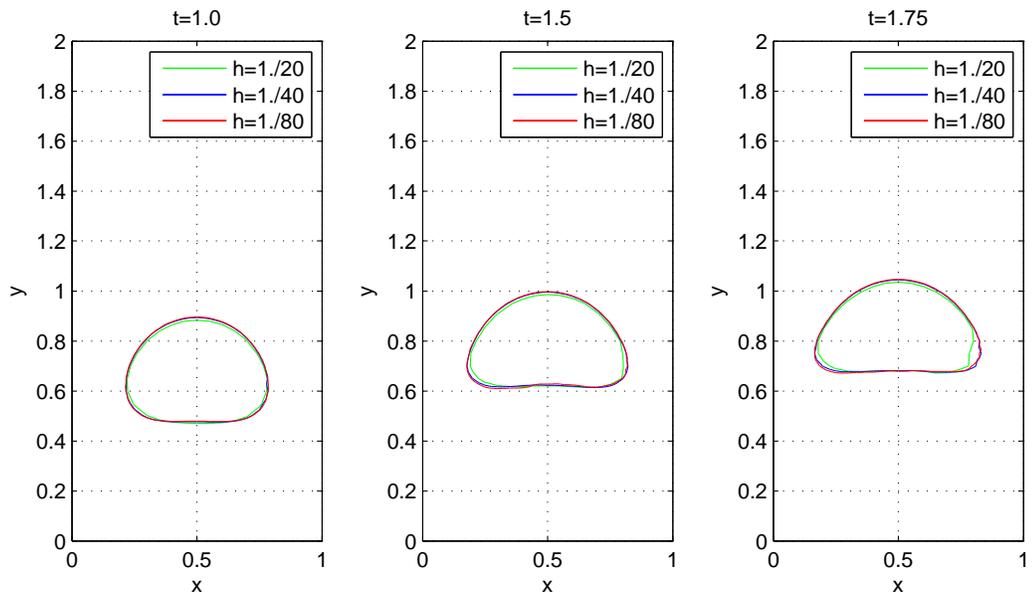


Figure 4.50: Bubble shapes under convergence tests from linearized three-step method with $\Delta t = 1./16h$ at $t = 1.0, 1.5, 1.75$. The problem was solved in non-dimensional variables, see Remark 3.14.

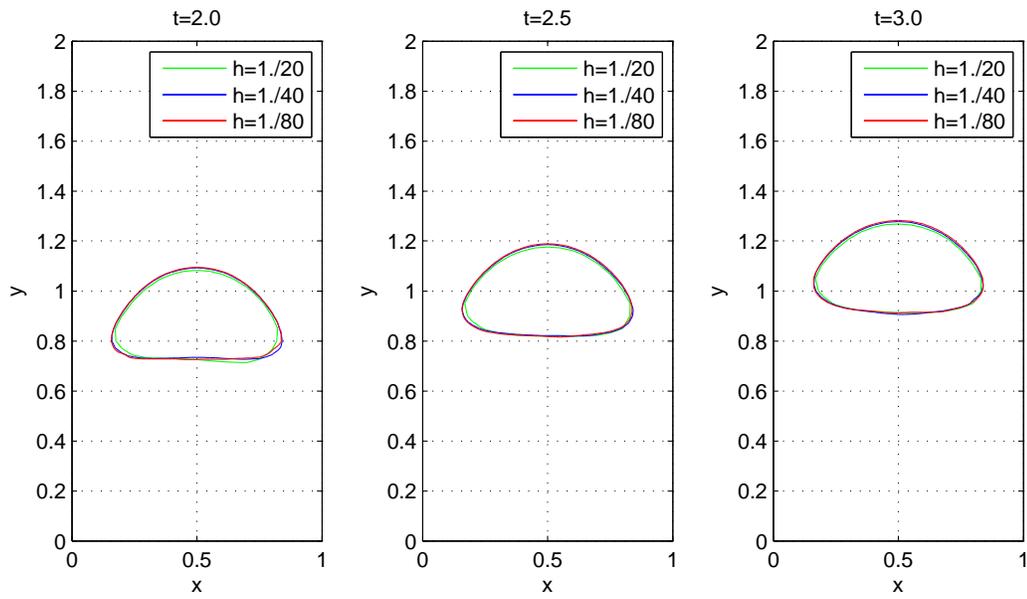


Figure 4.51: Bubble shapes under convergence tests from linearized three-step method with $\Delta t = 1./16h$ at $t = 2.0, 2.5, 3.0$. The problem was solved in non-dimensional variables, see Remark 3.14.

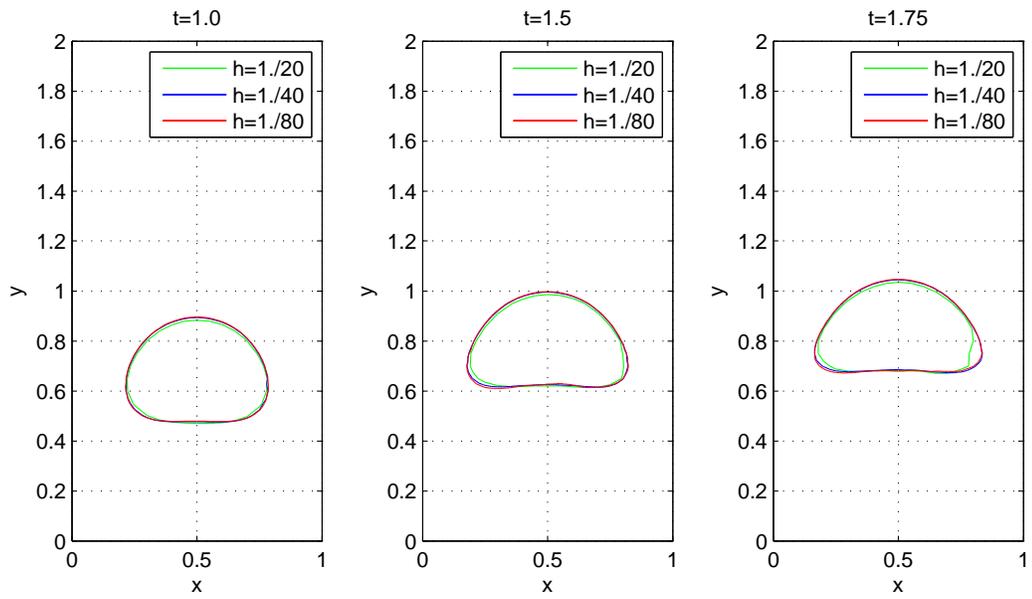


Figure 4.52: Bubble shapes under convergence tests from linearized two-step method with $\Delta t = 1./16h$ at $t = 1.0, 1.5, 1.75$. The problem was solved in non-dimensional variables, see Remark 3.14.

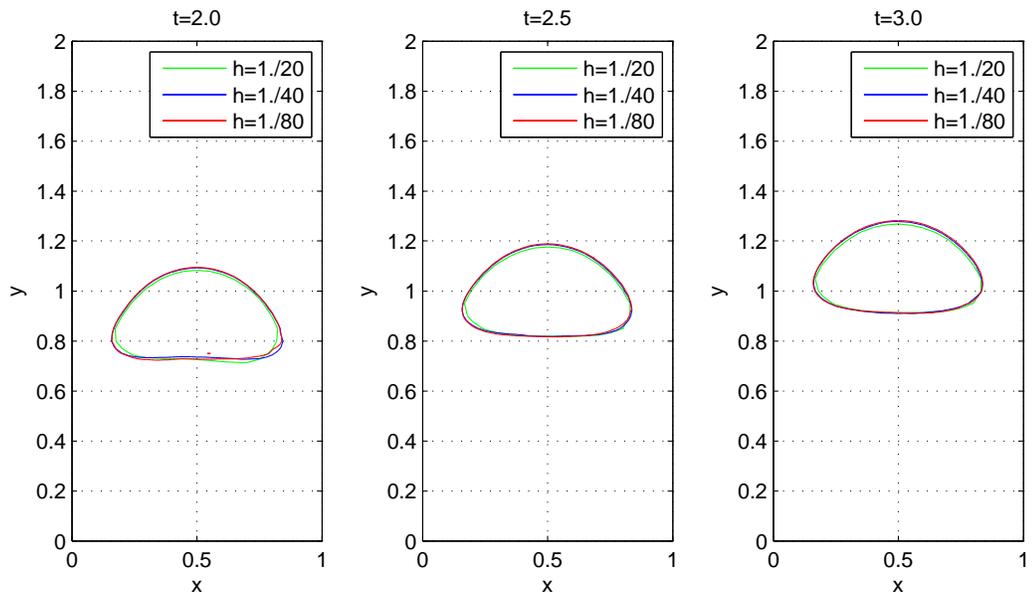


Figure 4.53: Bubble shapes under convergence tests from linearized two-step method with $\Delta t = 1./16h$ at $t = 2.0, 2.5, 3.0$. The problem was solved in non-dimensional variables, see Remark 3.14.

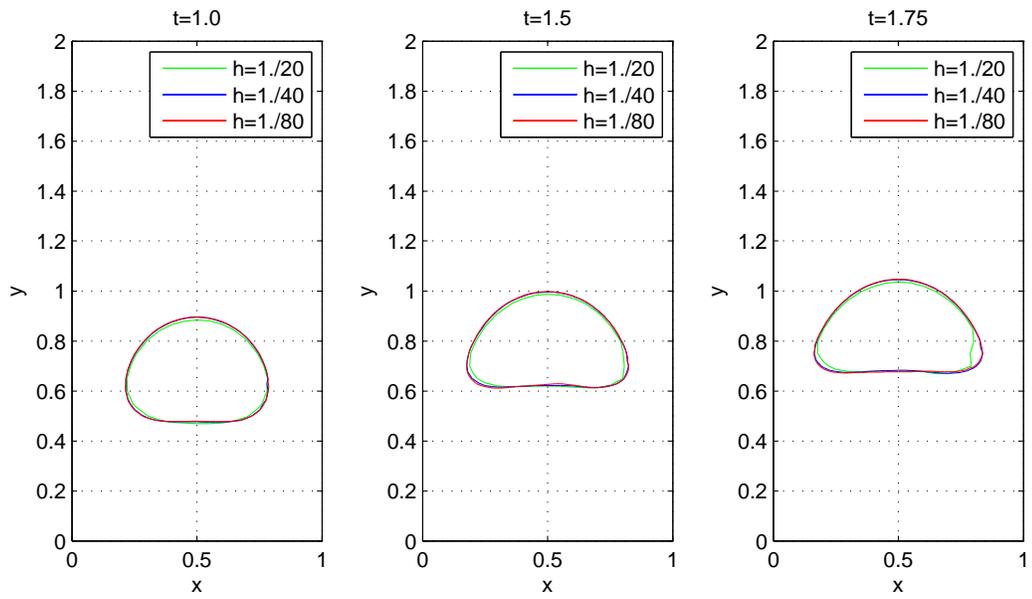


Figure 4.54: Bubble shapes under convergence tests from linearized Van Kan method with $\Delta t = 1./16h$ at $t = 1.0, 1.5, 1.75$. The problem was solved in non-dimensional variables, see Remark 3.14.

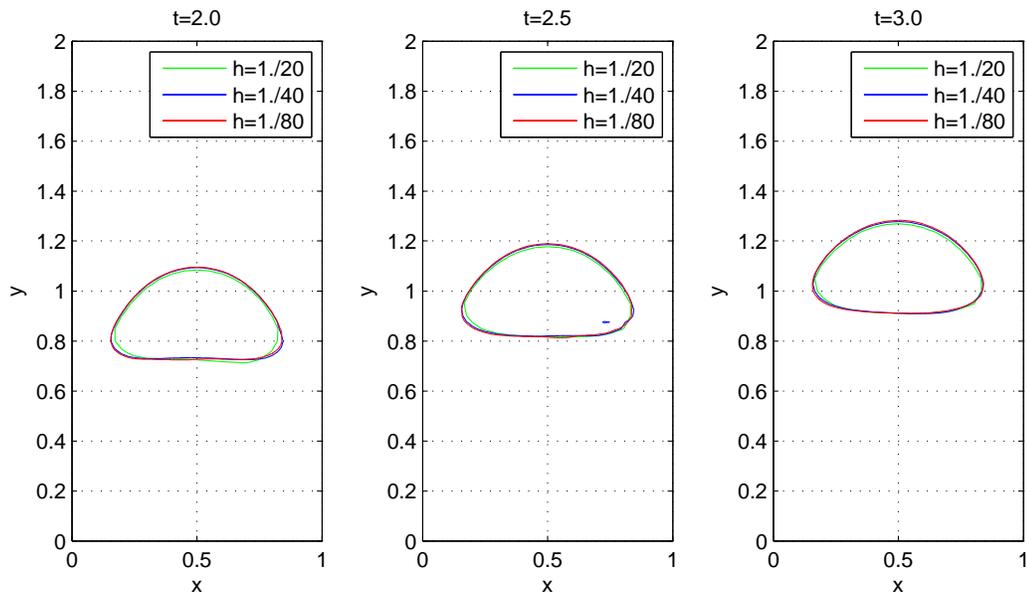


Figure 4.55: Bubble shapes under convergence tests from linearized Van Kan method with $\Delta t = 1./16h$ at $t = 2.0, 2.5, 3.0$. The problem was solved in non-dimensional variables, see Remark 3.14.

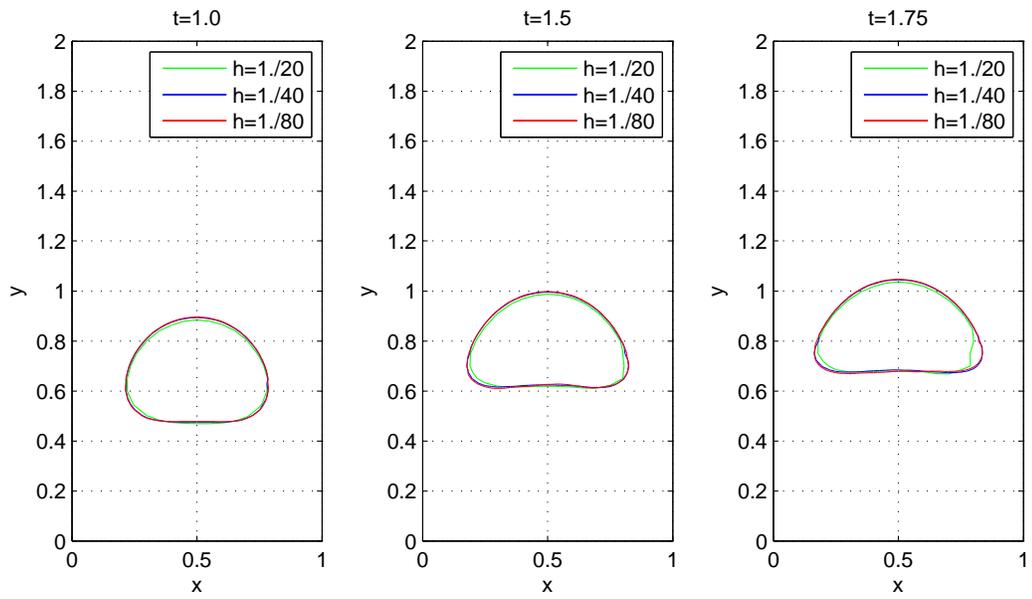


Figure 4.56: Bubble shapes under convergence tests from non-linearized three-step method with $\Delta t = 1./16h$ at $t = 1.0, 1.5, 1.75$. The problem was solved in non-dimensional variables, see Remark 3.14.

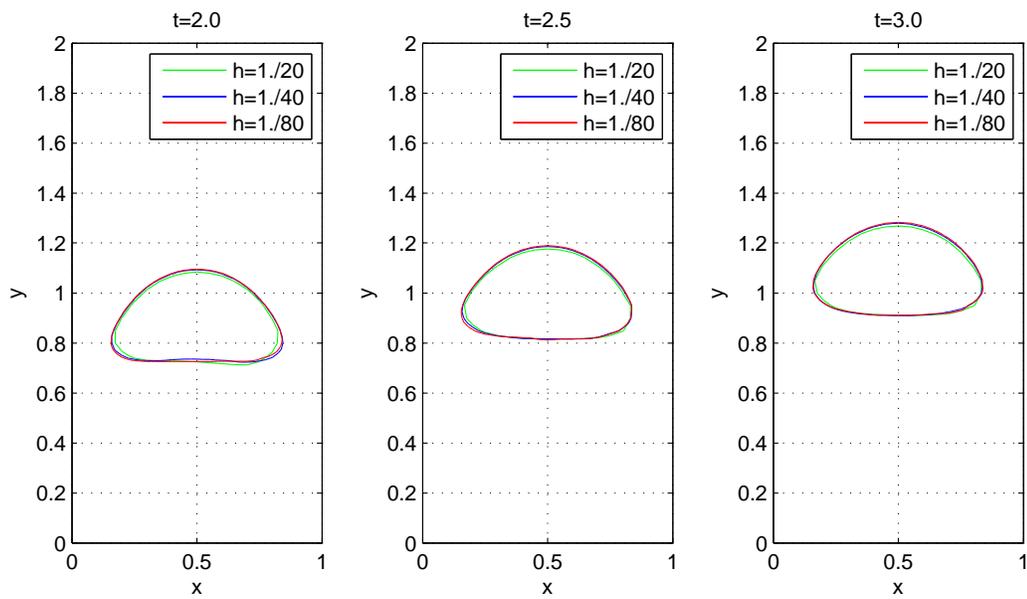


Figure 4.57: Bubble shapes under convergence tests from non-linearized three-step method with $\Delta t = 1./16h$ at $t = 2.0, 2.5, 3.0$. The problem was solved in non-dimensional variables, see Remark 3.14.

It is more interesting to validate our numerical schemes through the numerical quantities. The center of mass, the rise velocity and the circularity from all schemes are listed in Figure (4.58) to (4.69).

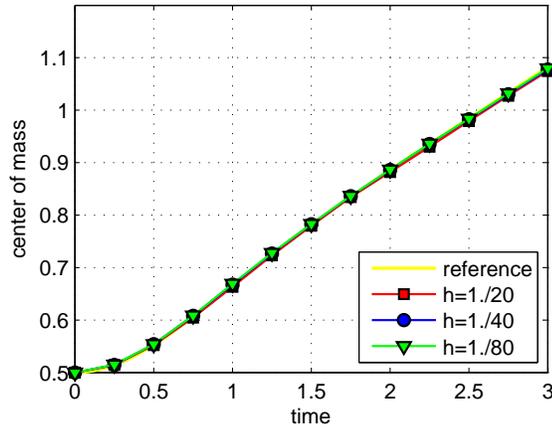


Figure 4.58: Center of mass from linearized three-step method under grid refinement with $\Delta t = 1./16h$ (reference solutions are obtained from TP2D code). The problem was solved in non-dimensional variables, see Remark 3.14.

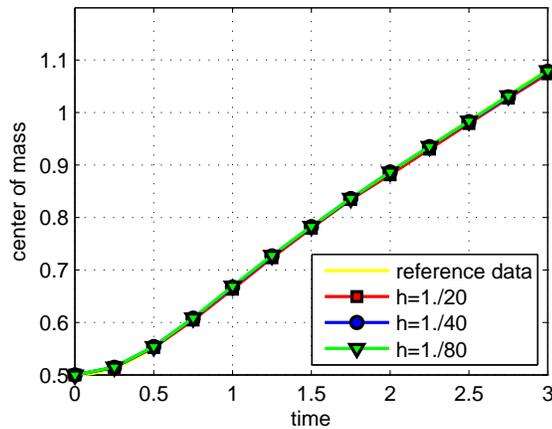


Figure 4.59: Center of mass from linearized two-step method under grid refinement with $\Delta t = 1./16h$ (reference solutions are obtained from TP2D code). The problem was solved in non-dimensional variables, see Remark 3.14.

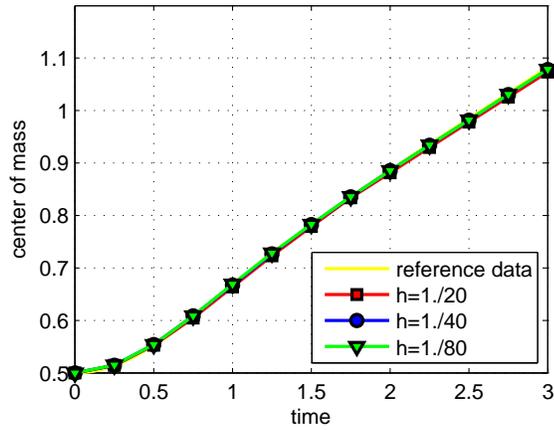


Figure 4.60: Center of mass from linearized Van Kan method under grid refinement with $\Delta t = 1./16h$ (reference solutions are obtained from TP2D code). The problem was solved in non-dimensional variables, see Remark 3.14.

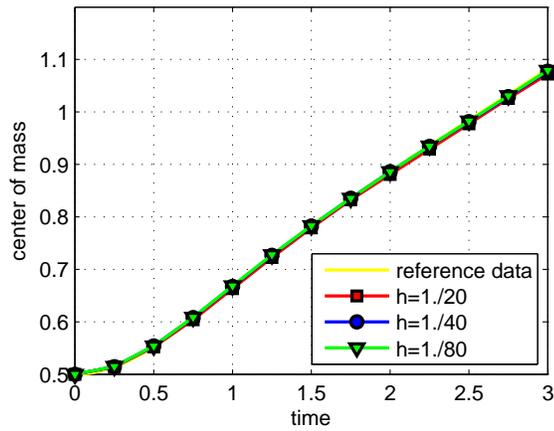


Figure 4.61: Center of mass from non-linearized three-step method under grid refinement with $\Delta t = 1./16h$ (reference solutions are obtained from TP2D code). The problem was solved in non-dimensional variables, see Remark 3.14.

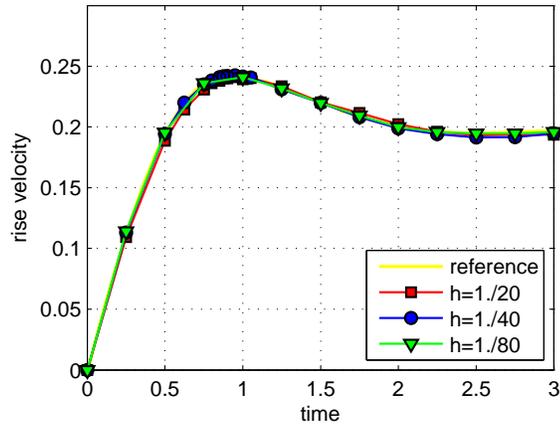


Figure 4.62: Rise velocity from linearized three-step method under grid refinement with $\Delta t = 1./16h$ (reference solutions are obtained from TP2D code). The problem was solved in non-dimensional variables, see Remark 3.14.

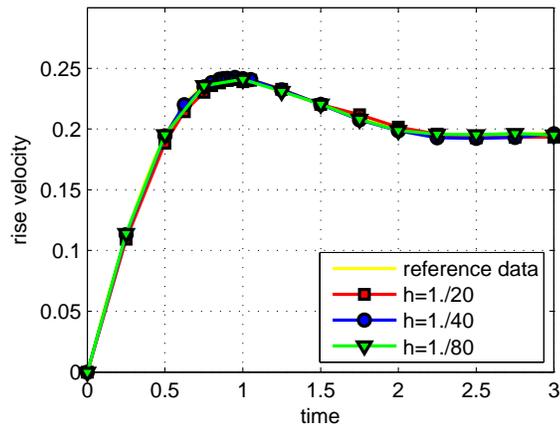


Figure 4.63: Rise velocity from linearized two-step method under grid refinement with $\Delta t = 1./16h$ (reference solutions are obtained from TP2D code). The problem was solved in non-dimensional variables, see Remark 3.14.

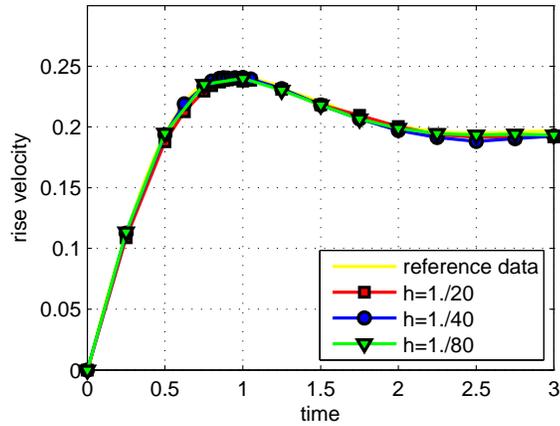


Figure 4.64: Rise velocity from linearized Van Kan method under grid refinement with $\Delta t = 1./16h$ (reference solutions are obtained from TP2D code). The problem was solved in non-dimensional variables, see Remark 3.14.

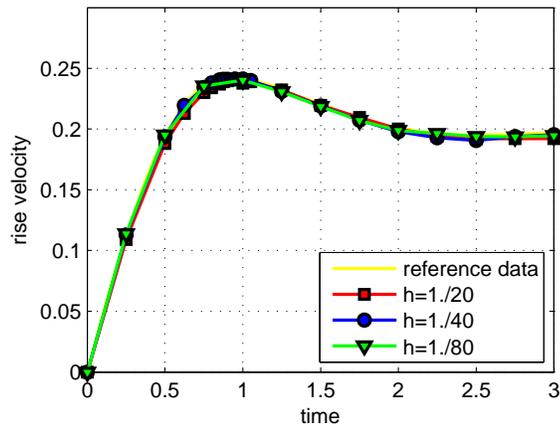


Figure 4.65: Rise velocity from non-linearized three-step method under grid refinement with $\Delta t = 1./16h$ (reference solutions are obtained from TP2D code). The problem was solved in non-dimensional variables, see Remark 3.14.

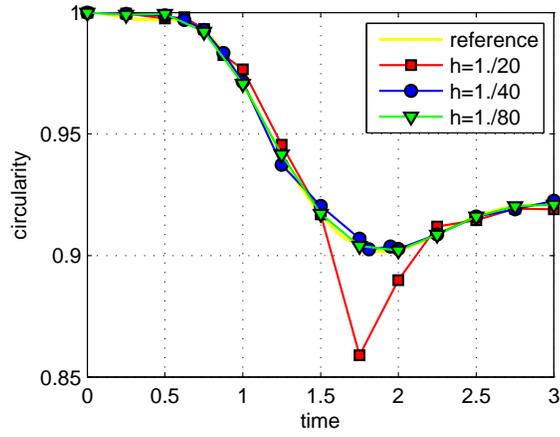


Figure 4.66: Circularity from linearized three-step method under grid refinement with $\Delta t = 1./16h$ (reference solutions are obtained from TP2D code). The problem was solved in non-dimensional variables, see Remark 3.14.

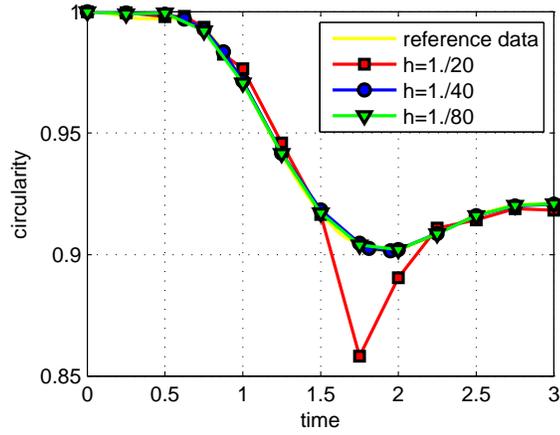


Figure 4.67: Circularity from linearized two-step method under grid refinement with $\Delta t = 1./16h$ (reference solutions are obtained from TP2D code). The problem was solved in non-dimensional variables, see Remark 3.14.

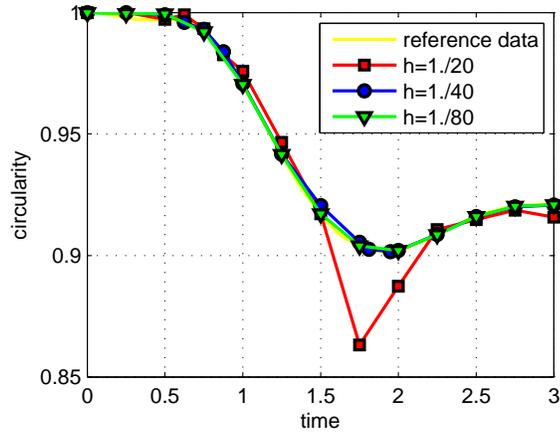


Figure 4.68: Circularity from linearized Van Kan method under grid refinement with $\Delta t = 1./16h$ (reference solutions are obtained from TP2D code). The problem was solved in non-dimensional variables, see Remark 3.14.

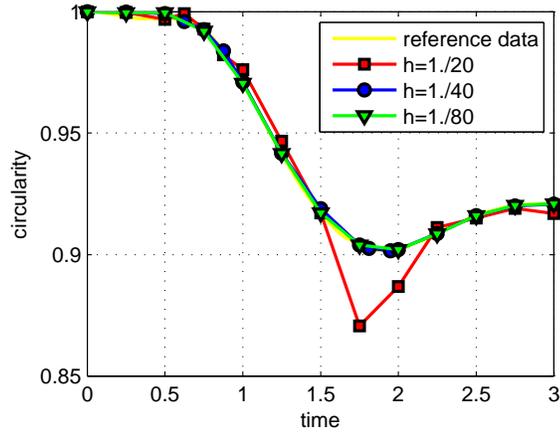


Figure 4.69: Circularity from non-linearized three-step method under grid refinement with $\Delta t = 1./16h$ (reference solutions are obtained from TP2D code). The problem was solved in non-dimensional variables, see Remark 3.14.

The center of mass and the rise velocity converge nicely for all schemes. Besides, the center of mass and the rise velocity at time $t = 3.0$ are useful parameters to further quantify the accuracy of our numerical schemes. These results are listed in Table (4.10) to (4.13). All results converge to the reference data in a good rate. The linearized three-step method

is more accurate in the center of mass towards the end of the simulation, and the linearized two-step method is more precise in the rise velocity.

Table 4.10: Linearized three-step method

$1/h$	20	40	80	reference
center of mass	1.0745	1.0776	1.0801	1.0818
rise velocity	0.1938	0.1947	0.1958	0.1970

Table 4.11: Linearized two-step method

$1/h$	20	40	80	reference
center of mass	1.0738	1.0785	1.0798	1.0818
rise velocity	0.1934	0.1951	0.1961	0.1970

Table 4.12: Linearized Van Kan method

$1/h$	20	40	80	reference
center of mass	1.0736	1.0783	1.0786	1.0818
rise velocity	0.1918	0.1927	0.1934	0.1970

Table 4.13: Non-linearized three-step method

$1/h$	20	40	80	reference
center of mass	1.0725	1.0776	1.0791	1.0818
rise velocity	0.1923	0.1946	0.1953	0.1970

The circularity is a challenging parameter due to the calculations of the perimeter of the bubble at different time. We used a linear approximation to capture the bubble locations.

At $t = 1.75$, the results from the coarser mesh with mesh size $h = 1./20$ lose certain masses, and this is why we can observe some deviations from the reference data between time $t = 1.5$ to $t = 2.0$. For the finer mesh sizes $h = 1./40$ and $h = 1./80$, the results match the reference data.

Results from COMSOL Multiphysics

Finding a reference data for the two-phase problem is not an easy task, so we complete the benchmark test using a standard commercial software named COMSOL Multiphysics to further verify our reference data. Our rising bubble test can be implemented utilizing the Chemical engineering module in COMSOL. In COMSOL, the Level Set method transport equation is solved using some stabilization techniques. Re-initialization is carried out at the same time by introducing the re-initialization parameters on the right-hand side of the transport equation. The discontinuous coefficients, the interface normals, and the surface tension forces are approximated using the resulting Level Set function. The Lagrange P2-P1 FEM pairs were used to approximate the velocity and the pressure field. Figure (4.70) compares the interface shape at the final time $t = 3$ from COMSOL with our reference data. The experiment was performed using mesh size $h = 1./40$, in 13722 elements with 90037 degrees of freedom.

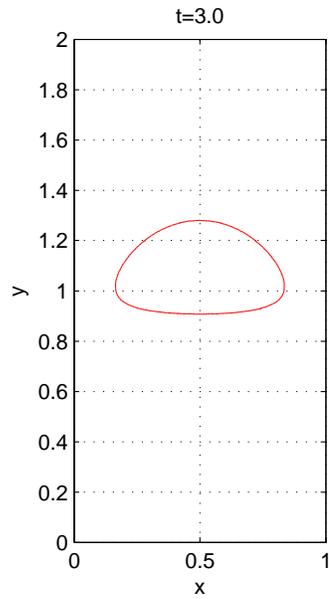


Figure 4.70: Babble shape at $t = 3.0$ with $\Delta t = 1./16h$ from COMSOL Multiphysics (reference solutions are obtained from TP2D code). The problem was solved in non-dimensional variables, see Remark 3.14.

The time evolutions of the interface shapes at different time are listed in Figure (4.71) and (4.72). The center of mass and the rise velocity are reported in Figure (4.73) and Figure (4.74). These two quantities also match the reference data.

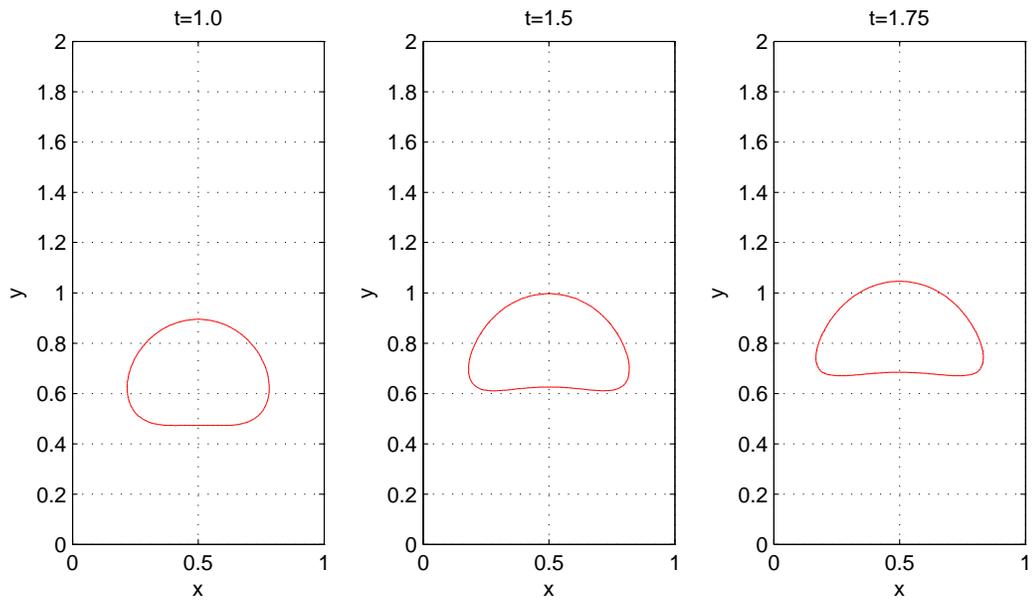


Figure 4.71: Time evolutions of interface shapes with $\Delta t = 1./16h$ at $t = 1.0, 1.5, 1.75$ from COMSOL Multiphysics. The problem was solved in non-dimensional variables, see Remark 3.14.

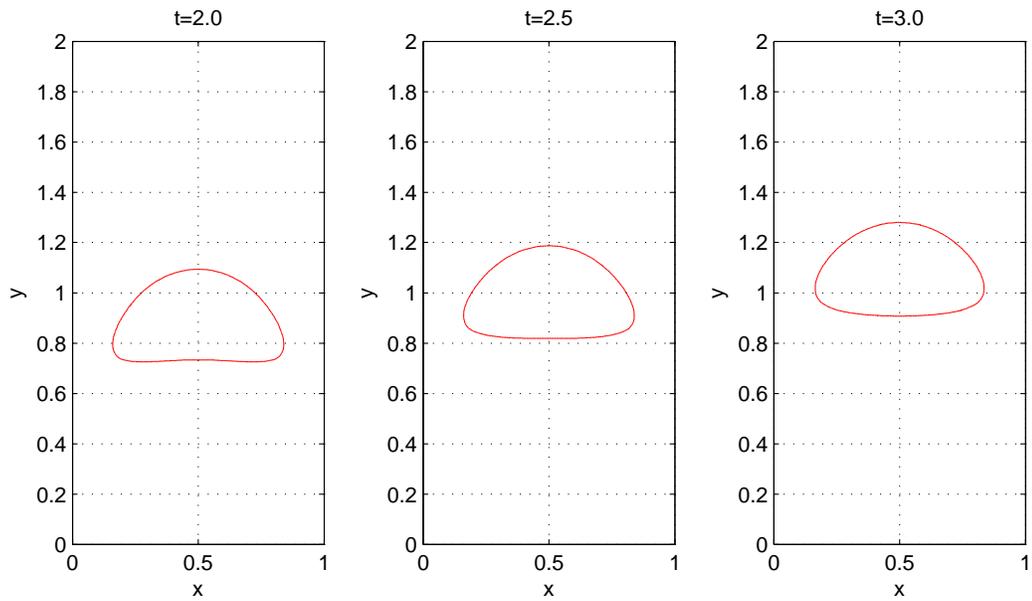


Figure 4.72: Time evolutions of interface shapes with $\Delta t = 1./16h$ at $t = 2.0, 2.5, 3.0$ from COMSOL Multiphysics. The problem was solved in non-dimensional variables, see Remark 3.14.

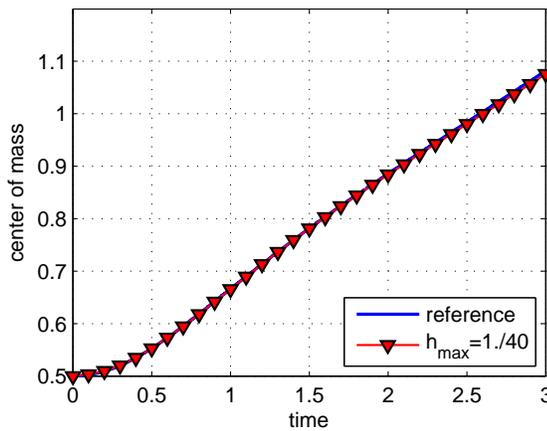


Figure 4.73: Center of mass from COMSOL Multiphysics with $\Delta t = 1./16h$ (reference solutions are obtained from TP2D code). The problem was solved in non-dimensional variables, see Remark 3.14.

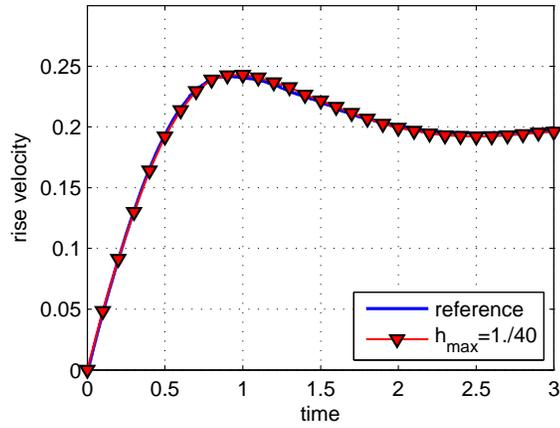


Figure 4.74: Rise velocity from COMSOL Multiphysics with $\Delta t = 1./16h$ (reference solutions are obtained from TP2D code). The problem was solved in non-dimensional variables, see Remark 3.14.

The center of mass and the rise velocity at the final time $t = 3.0$ are 1.0757 and 0.1962 respectively, and they also coincide with the reference data.

Chapter 5

Conclusions

In this dissertation, the numerical schemes to simulate the immiscible two-phase fluid flows have been discussed. Four solvers based on the operating splitting schemes coupled with the Level Set method for the interface modeling have been tested and calibrated. The velocity and the pressure were approximated using the P2-P1 Finite Elements. The P1 Finite Element basis functions were used in the Level Set transport equation.

A Volume Fraction method has been developed in order to evaluate the discontinuous coefficients. Compared with the traditional regularization techniques, this new method does not require the approximations of the Dirac Delta functions. The corresponding integrals, for example, the mass matrix can be computed exactly due to the presence of the P1 basis functions in the Volume Fraction method. The performance for the new method was tested in form of the error analysis of integrations for a Characteristic function, a second order, a fourth order, and an exponential function.

The surface tension force has been implemented in a semi-implicit way, where larger time steps can be taken compared with the explicit methods. In addition, numerical

approximations of the interface normals, the curvature, as well as the Dirac Delta functions are not necessary by taking advantage of the tangential derivatives. A recently developed minimization-based re-initialization approach has been adopted into the two-phase flow solvers, where the standard FEM is implemented again to solve the resulting elliptic PDEs. Thus the zero Level Set has been strictly preserved by doing so.

The solvers have been validated in several benchmark problems. The idea is, starting with convection diffusion problem with one-phase flow, we gradually add back the gradient of pressure and divergence free terms. After solvers worked for the one-phase Navier Stokes equations, the surface tension effect and the re-initialization step have been implemented. The two-phase flow solvers have been built taking into account the discontinuous coefficients. Picture norms and the numerical quantities have been checked in each benchmark problem. The solvers have been validated finally using a two-phase rising bubble benchmark problem. The solutions have been compared with the results from other research groups. The results from COMSOL Multiphysics have been included to verify our reference data. This standard commercial tool further validated our solvers in simulating the two-phase flow problem.

Bibliography

- [1] I. Babuska. Error bounds for Finite Element method. *Numer. Math.*, 16(4):322–333, 1971.
- [2] Eberhard Bänsch. Finite Element discretization of the Navier-Stokes equations with a free capillary surface. *Numerische Mathematik*, 88(2):203–235, 2001.
- [3] Brill Beggs. A study of two phase flow in inclined pipes. *Journal of Petroleum Technology*, 25(5):39–48, 1973.
- [4] F. Brezzi. On the existence, uniqueness and approximation of saddle point problems arising from Lagrangian multipliers. *RAIRO Anal. Numer.*, 8(R2):129–151, 1974.
- [5] D. Chisholm. *Two-phase Flow in Pipelines and Heat Exchanges*. Bath, UK: Pitman Press Limited, 1983.
- [6] A. J. Chorin. Numerical solution of the Navier-Stokes equations. *Math. Comput*, 22(104):745–762, 1968.
- [7] A. J. Chorin. On the convergence of discrete approximations to the Navier-Stokes equations. *Math. Comput*, 23(106):341–353, 1969.
- [8] P. K. Kundu, I. M. Cohen. *Fluid Mechanics*. Academic, California, 2004.
- [9] R. Elias, M. Martins, A. Coutinho. Simple Finite Element-based computation of distance functions in unstructured grids. *International Journal for Numerical Methods in Engineering*, 72(9):1095–1110, 2007.
- [10] T.A. Davis. *Direct Methods for Sparse Linear Systems*. Society for Industrial and Applied Mathematics, Philadelphia, 2006.
- [11] J. Donea. A Taylor-Galerkin method for convective transport problems. *International Journal for Numerical Methods in Engineering*, 20(24):101–120, 1984.
- [12] D. N. Arnold, F. Brezzi, M. Fortin. A stable Finite Element for the Stokes equations. *Calcolo*, 21(4):337–344, 1984.

- [13] C. Li, C. Xu, C. Gui, M. D. Fox. Distance regularized Level Set evolution and its application to image segmentation. *IEEE Trans. Image Process.*, 19(12):3243–3254, 2010.
- [14] T. J. R. Hughes, L.P. Franca. A new Finite Element formulation for computational fluid dynamics. VII. the Stokes problem with various well-posed boundary conditions: Symmetric formulations that converge for all velocity pressure spaces. *Comput. Methods Appl. Mech. Eng.*, 65(1):85–96, 1987.
- [15] P. M. Gresho. On the theory of semi-implicit projection methods for viscous incompressible flow and its implementation via a Finite Element method that also introduces a nearly consistent mass matrix. Part 1: Theory. *International Journal for Numerical Methods in Fluids*, 11(5):587–620, 1990.
- [16] Sven Groß. Numerical methods for three-dimensional incompressible two-phase flow problems. *PhD thesis, Rheinisch-Westfälische Technische Hochschule Aachen University*, 2008.
- [17] C. Taylor, P. Hood. A numerical solution of the Navier-Stokes equations using the Finite Element technique. *Comput. and Fluids*, 1(1):73–100, 1973.
- [18] J. Donea, A. Huerta. *Finite Element Methods for Flow Problems*. John Wiley and Sons, Chichester, 2003.
- [19] S. Hysing. A new implicit surface tension implementation for interfacial flows. *International Journal for Numerical Methods in Fluids*, 51(6):659–672, 2006.
- [20] S. Hysing. Numerical simulation of immiscible fluids with FEM Level Set techniques. *PhD thesis, Dem Fachbereich Mathematik der Universität Dortmund*, 2007.
- [21] Christopher Basting, Dmitri Kuzmin. A minimization-based Finite Element formulation for interface-preserving Level Set reinitialization. *Computing*, 95(1):13–25, 2013.
- [22] T. J. R. Hughes, W. K. Liu, T. K. Zimmermann. Lagrangian-Eulerian Finite Element formulation for incompressible viscous flows. *U.S.-Japan Seminar on Interdisciplinary Finite Element Analysis*, 29(3), 1978.
- [23] O. A. Ladyzhenskaya. *The Mathematical Theory of Viscous Incompressible Flow*. Gordon and Breach Science, New York, 1969.
- [24] H. Lamb. *Hydrodynamics*. Cambridge University Press, Cambridge, 2005.
- [25] L. Chen, S. V. Garimella, J. A. Reizes, E. Leonardi. The development of a bubble rising in a viscous fluid. *Journal of Fluid Mechanics*, 387:61–96, 1999.
- [26] R. Glowinski, P. LeTallec. *Augmented Lagrangian and Operator-Splitting Methods in Nonlinear Mechanics*. Society for Industrial and Applied Mathematics, Philadelphia, 1989.

- [27] R. Courant, K. Friedrichs, H. Lewy. On the partial difference equations of mathematical physics. *Mathematische Annalen (in German)*, 100(1):32–74, 1928.
- [28] G. I. Marchuk. *Methods of Numerical Mathematics*. Second edition, Springer-Verlag, New York, 1982.
- [29] G. I. Marchuk. *Splitting and Alternating Direction Methods*, volume 1 of *Handbook of Numerical Analysis*. Elsevier, 1990.
- [30] R.W. Lockhart, R.C. Martinelli. Proposed correlation of data for isothermal two-phase, two-component flow in pipes. *Chemical Engineering Progress*, 45(1):39–48, 1949.
- [31] V. John, G. Matthies. Higher-order Finite Element discretizations in a benchmark problem for incompressible flows. *International Journal for Numerical Methods in Fluids*, 37:885–903, 2011.
- [32] Jeffrey Slotnick, Abdollah Khodadoust, Juan Alonso, David Darmofal, William Gropp, Elizabeth Lurie, Dimitri Mavriplis. CFD vision 2030 study: A path to revolutionary computational aerosciences. Technical report, National Aeronautics and Space Administration, March 2014.
- [33] C. E. Norman, M. J. Miksis. Dynamics of a gas bubble rising in an inclined channel at finite reynolds number. *Physics of Fluids*, 17(2):22–102, 2005.
- [34] G. Nabh. On high order methods for the stationary incompressible Navier-Stokes equations. *PhD thesis, University of Heidelberg*, 1998.
- [35] T. Chen, P. D. Mineev, K. Nandakumar. A projection scheme for incompressible multiphase flow using adaptive Eulerian grid. *Int. J. Num. Meth. Fluids*, 45(1):1–19, 2001.
- [36] C. W. Hirt, B. D. Nichols. Volume of Fluid (VOF) methods for the dynamics of free boundaries. *Journal of Computational Physics*, 39(1):201–225, 1981.
- [37] Mark Sussman, Peter Smereka, Stanley Osher. A Level Set approach for computing solutions to incompressible two-phase flow. *Journal of Computational Physics*, 114(1):146 – 159, 1994.
- [38] Y.C. Chang, T.Y. Hou, B. Merriman, S. Osher. A Level Set formulation of Eulerian interface capturing methods for incompressible fluid flows. *Journal of Computational Physics*, 124(2):449–464, 1996.
- [39] N. Parolini. Computational fluid dynamics for naval engineering problems. *PhD thesis, École Polytechnique Fédérale de Lausanne*, 2004.
- [40] C. S. Peskin. Numerical analysis of blood flow in the heart. *Journal of Computational Physics*, 25(3):220–252, 1977.

- [41] M. Bercovier, O. Pironneau. Error estimates for Finite Element solution of the Stokes problem in the primitive variables. *Numer. Math.*, 33(2):211–224, 1979.
- [42] M. Sussman, E. G. Puckett. A coupled Level Set and Volume of Fluid method for computing 3d and axisymmetric incompressible two-phase flows. *Journal of Computational Physics*, 162(2):301–337, 2000.
- [43] J.-L. Guermond, L. Quartapelle. On stability and convergence of projection methods based on pressure Poisson equation. *International Journal for Numerical Methods in Fluids*, 26(9):1039–1053, 1998a.
- [44] J.-L. Guermond, L. Quartapelle. On the approximation of the unsteady Navier-Stokes equations by Finite Element projection methods. *Numer. Math.*, 80(2):207–238, 1998b.
- [45] M. Crouzeix, P.-A. Raviart. Conforming and nonconforming Finite Element methods for solving the stationary Stokes equations i. *ESAIM: Mathematical Modelling and Numerical Analysis*, 7(R3):33–75, 1973.
- [46] V. Girault, P.-A. Raviart. *Finite Element Methods for Navier-Stokes Equations*. Springer-Verlag, Berlin, 1986.
- [47] J.N. Reddy. *An Introduction to the Finite Element Method (Third ed.)*. McGraw-Hill, 2005.
- [48] P. M. Gresho, R.L. Sani. On pressure boundary conditions for the incompressible Navier-Stokes equations. *International Journal for Numerical Methods in Fluids*, 7(10):1111–1145, 1987.
- [49] J. A. Sethian. A fast marching Level Set method for monotonically advancing fronts. *Proceedings of the National Academy of Sciences*, 93(4):1591–1595, 1996.
- [50] S. Osher, J.A. Sethian. Fronts propagating with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulations. *Journal of Computational Physics*, 79(1):12–49, 1988.
- [51] M. Sussman, P. Smereka. Axisymmetric free boundary problems. *Journal of Fluid Mechanics*, 341:269–294, 1997.
- [52] Anton Smolianski. Numerical modeling of two-fluid interfacial flows. *PhD thesis, University of Jyväskylä*, 2001.
- [53] S.Pissanetsky. *Sparse Matrix Technology*. Academic Press, 1984.
- [54] R. Temam. Surl’s approximation de la solutions des equations de Navier-Stokes par la methoded des pas fractionnaires. *Arch. Ration. Mech. Anal*, 33(2):377–385, 1969.
- [55] R. Temam. Remark on the pressure boundary condition for the projection method. *Theory of Computational Fluid Dynamics.*, 3(3):181–184, 1991.

- [56] R. Temam. *Navier-Stokes Equations. Theory and Numerical Analysis*. AMS Chelsea Publishing, Providence, RI. Corrected reprint for the 1984 edition, 2004.
- [57] Anna-Karin Tornberg. Interface tracking methods with application to multiphase flows. *PhD thesis, Royal Institute of Technology, Department of Numerical Analysis and Computing Science*, 2000.
- [58] S. O. Unverdi, G. Tryggvason. A front-tracking method for viscous, incompressible, multi-fluid flows. *Journal of Computational Physics*, 100(1):25–37, 1992.
- [59] B. Engquist, A.-K. Tornberg, R. Tsai. Discretization of Dirac Delta functions in Level Set methods. *Journal of Computational Physics*, 207(1):28 – 51, 2005.
- [60] S. Hysing, S. Turek. The Eikonal equation: numerical efficiency vs. algorithmic complexity on quadrilateral grids. In *Proceedings of Algoritmy, Conference on Scientific Computing*, pages 22–31, 2005.
- [61] A. Quarteroni, A. Valli. *Numerical Approximation of Partial Differential Equations*. Springer-Verlag, Berlin, 1994.
- [62] H. Wadell. Sphericity and roundness of rock particles. *Journal of Geology*, 41(3):310–331, 1933.
- [63] R. Clift, J. R. Grace, M. E. Weber. *Bubbles, Drops and Particles*. Academic Press: New York, 1978.
- [64] F. H. Harlow, J. E. Welch. Numerical calculation of time-dependent viscous incompressible flow of fluid with a free surface. *Physics of Fluids*, 8(12):2182–2189, 1965.
- [65] F. H. Harlow, J. E. Welch. *SLIC (Simple Line Interface Calculation)*, volume 59 of *Lecture Notes in Physics*. Springer, New York, 1976.
- [66] N. N. Yanenko. *The Method of Fractional Steps. The Solution of Problems of Mathematical Physics in Several Variables*. Springer-Verlag, Berlin, 1971.
- [67] B. Lafaurie, C. Nardone, R. Scardovelli, S. Zaleski, G. Zanetti. Modelling merging and fragmentation in multiphase flows with SURFER. *Journal of Computational Physics*, 113(1):134–147, 1994.
- [68] J.U. Brackbill, D.B. Kothe, C. Zemach. A continuum method for modeling surface tension. *Journal of Computational Physics*, 100(2):335–354, 1992.
- [69] Y. Tsai, L. Cheng, S. Osher, H. Zhao. Fast sweeping algorithms for a class of Hamilton-Jacobi equations. *SIAM Journal on Numerical Analysis*, 41(2):673–694, 2003.
- [70] J. C. Heinrich, R.S. Marshall, O.C. Zienkiewicz. *Penalty Function Solution of Coupled Convective and Conductive Heat Transfer*. Numerical Methods in Laminar and Turbulent Flows, Pentech Press, 1978.