Diffuse Solid-Fluid Interface Method for Dispersed Multiphase Flows

by

Tanyakarn Treeratanaphitak

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Chemical Engineering

Waterloo, Ontario, Canada, 2018

Examining Committee Membership

The following faculty served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

External Examiner:	Li Xi Assistant Professor Dept. of Chemical Engineering McMaster University
Supervisor:	Nasser Mohieddin Abukhdeir Associate Professor Dept. of Chemical Engineering University of Waterloo
Internal Members:	Hector Budman Professor Dept. of Chemical Engineering University of Waterloo
	Jeff Gostick Associate Professor Dept. of Chemical Engineering University of Waterloo
Internal-External Member:	Fue-Sang Lien Professor Dept. of Mechanical and Mechatronics Engineering University of Waterloo

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

Section 3.1.2 in Chapter 3 and Chapters 4 and 5 have been incorporated in a paper that has been submitted for publication, T. Treeratanaphitak and N. M. Abukhdeir. Phase-bounded finite element method for two-fluid incompressible flow systems. *Int. J. Multiphase Flow* (Submitted July 3, 2018).

Abstract

Industrial chemical engineering processes such as bubble columns, reactors and separators involve multiphase flows of two or more fluids. In order to improve the design and operation of these processes, an understanding of their multiphase hydrodynamics is essential. An emergent tool in studying multiphase flow systems that is becoming readily accessible to researchers is computational fluid dynamics (CFD) simulation. CFD simulations of multiphase flow systems enable researchers to explore the effect of different combinations of operating conditions and designs on pressure drop, separation efficiency, and heat and mass transfer without the cost and safety issues incurred by experimental design and pilot studies. Consequently, CFD simulations are increasingly relevant for the design and optimization of chemical process equipment. The multiphase hydrodynamic model that is often used to study chemical engineering processes is the two-fluid (Euler-Euler) model. In this model, the fluids are treated as inter-penetrating continua and fluid phase fractions are used to describe the average spatial composition of the multiphase fluid.

Generally, the physical boundaries (e.g. vessel walls, reactor internals, *etc.*) in numerical simulations using the two-fluid model are defined by the mesh or grid, i.e. the mesh/grid boundaries correspond to an approximation of the physical boundaries of the system. The resulting conformal mesh/grid could potentially contain a large number of skewed elements, which is undesirable in numerical simulations. One approach to address this issue involves approximation of solid boundaries using a diffuse solid-fluid interface approximation. This approach allows for a structured mesh to be used while still capturing the desired solid-fluid boundaries. The diffuse-interface method also allows for the simulation of moving boundaries without the need for manipulation of the underlying mesh/grid or interpolation of boundary variables to the nearest node. This allows for the geometry of the domain of interest (i.e. process equipment) to be easily modified during the process of simulation-assisted design and optimization.

In the two-fluid model, phase fractions are used to describe the composition of the mixture and are bounded quantities. Consequently, numerical solution methods used in simulations must preserve boundedness for accuracy and physical fidelity. Firstly, a phase-bounded numerical method for the two-fluid model is developed in which phase fraction

inequality constraints are imposed through the use of an implicit variational nonlinear inequality solver. The numerical method is verified and compared to an established explicit numerical method. The effect of using separate phasic pressure fields as opposed to the commonly used single-pressure assumption is also found to be non-negligible in dilute dispersed flows (less than 3% gas fraction).

Subsequently, the phase-bounded numerical method is extended to support a diffuse-interface method for the imposition of solid-fluid boundaries. The diffuse-interface is used to define physical boundaries and boundary conditions are imposed by blending conservation equations from the two-fluid model with the solid boundary condition. Simulations of two-dimensional channel flow and flow past a stationary cylinder are used to validate the diffuse-interface method. This is achieved by comparing the bubble plume width and time evolution of the overall gas hold-up from the diffuse-interface simulations with results obtained using boundary-conformal meshes. The results from the channel flow simulations are found to be in agreement with the boundary-conformal mesh solution when the interface width is sufficiently small. In the case of flow past a stationary cylinder, similar flow features are observed in both diffuse-interface and reference simulations.

Acknowledgments

First and foremost, I would like to express my deepest gratitude to my family, for none of this would be possible without their encouragement and support.

I would like to thank my supervisor, Professor Nasser Mohieddin Abukhdeir the guidance, support and this opportunity. I would also like to thank my committee members, Professors Li Xi (McMaster University), Fue-Sang Lien, Hector Budman and Jeff Gostick for their feedback on this work and Professor Marios Ioannidis for his insightful comments.

Thank you to Amir Mowla and Dr. Mehrez Agnaou for all of the discussions that we have had over the course of my studies and for your support. A special thanks goes to HW, CCC, JJS, BF, CL and MS, for your longstanding friendship and unwavering support, and to members of the Thai Students Association for being a reliable sounding board. Thank you to past and present members of ARG/UW-CMRG/ComPhys for all of the engaging discussions (work-related or otherwise) throughout the years.

Lastly, I would like to thank Shell Canada, Natural Sciences and Engineering Research Council of Canada (NSERC), University of Waterloo and Andrew and Margaret Stephens Graduate Scholarship in Chemical Engineering for their financial support and Compute Canada for the use of their computational facilities.

Table of Contents

List of Tables xvii

List of Figures xix

List of Symbols xxiii

1 Introduction 1

1.1 Research Motivation 1

1.2 Objectives 4

1.3 Structure of Thesis 5

2 Background 7

2.1 Gas-Liquid Hydrodynamics 7

2.1.1 Horizontal Flow 7

2.1.2 Vertical Flow 8

2.2 Two-Fluid Model 10

2.2.1 Mass Conservation 11

2.2.2 Momentum Conservation 12

2.2.3 Interfacial Forces 13

2.3 Numerical Method for Partial Differential Equations 16
2.3.1 Method of Weighted Residuals 16
2.3.2 Finite Element Method 17
2.3.3 Method of Lines 19
2.3.4 Solution Methods for Single-Phase Incompressible Navier-Stokes Equations 22

3 Literature Review **27**
3.1 Solution Methods for the Two-Fluid Model 27
3.1.1 Phase-Fraction Boundedness in the Two-Fluid Model 28
3.1.2 Well-Posedness of the Two-Fluid Model 31
3.2 Embedded Domain Methods for Multiphase Flows 33
3.2.1 Immersed Boundary Method 33
3.2.2 Diffuse-Interface Method 41

4 Finite Element Formulation for the Solution of the Two-Fluid Model **43**
4.1 Modified Incremental Pressure Correction Scheme 43
4.2 Weak Form of Governing Equations 47
4.3 Adaptive Time-Stepping 51

5 Phase-Bounded Finite Element Method for Two-Fluid Incompressible Flow Systems **53**
5.1 Methodology 53
5.1.1 Simulation Conditions 53
5.1.2 Numerical Methods 56
5.2 Results and Discussion 57

5.2.1 Effects of Phase Fraction Boundedness, $P_c = P_{int}$ 57
5.2.2 Effects of Phase Fraction Boundedness $P_c \neq P_{int}$ 62
5.3 Conclusions 63

6 Diffuse-Interface Method for Physical Boundaries **65**
6.1 Methodology 65
6.1.1 Numerical Method 67
6.1.2 Diffuse-Interface for Two-Fluid Model Equations 68
6.1.3 Simulation Conditions 72
6.2 Results and Discussion 75
6.2.1 Channel Flow 75
6.2.2 Flow Past a Cylinder 90
6.3 Conclusions 92

7 Conclusions and Recommendations for Future Work **97**
7.1 Conclusions 97
7.2 Recommendations for Future Work 98

References **101**

Appendix A Supporting Information **117**
A.1 Chapter 5 – Grid Convergence 117
A.2 Chapter 6 – Grid Convergence 117

List of Tables

4.1 Dimensionless groups 45

5.1 Physical properties 55

5.2 Initial and boundary conditions. 55

6.1 Initial and inlet conditions for gas-liquid channel flow with diffuse-interface. 74

6.2 Initial and inlet conditions for gas-liquid flow past a cylinder. 74

6.3 Bubble plume width at $y = 0.08\,\mathrm{m}$ from simulations using hyperbolic tangent diffuse-interface. 83

6.4 Bubble plume width at $y = 0.08\,\mathrm{m}$ from simulations using piece-wise cosine diffuse-interface. 89

List of Figures

2.1 Different flow regimes in gas-liquid flow in horizontal pipes [60]. 8

2.2 Flow regime as a function of gas superficial velocity and column diameter [15]. 9

2.3 Different flow regimes in cocurrent gas-liquid flow in vertical pipes [61]. . . 10

2.4 Different flow regimes in gas-liquid bubble columns [14]. 11

2.5 Linear piece-wise representation of y_a (solid line) and the corresponding basis functions (dashed lines). Figure adapted from Ref. [75]. 19

2.6 Quadrilateral elements and element shape functions [76]. 20

2.7 Method of lines solution [77]. 21

3.1 Multiphase simulation of a Diesel injector using the immersed boundary method [125]. 36

3.2 Comparison of free surface profiles at different times in Ref. [123]. – results from Ref. [123] □ reference simulation. 38

3.3 Free surface elevation from ship hydrodynamics simulation in Ref. [123] . . 38

3.4 Gas-liquid-solid rotating drum. (left) Simulation results (right) experimental results, dashed line indicate computed bed shape from simulations [132]. 40

3.5 Solid ball impacting into a liquid-liquid interface [54]. 42

5.1 Simulation domain. 54

5.2 Evolution of the phase fraction and liquid velocity streamline over time. Colors denote α_g. 58

5.3 Surface plot of (left) phase fraction, (center) gas velocity and (right) liquid velocity at $t = 1.72\,\mathrm{s}$ from (top) unbounded IPCS, (middle) bounded IPCS and (bottom) `twoPhaseEulerFoam`. 59

5.4 Evolution of $\min(\alpha_g)$ over time. 60

5.5 Two-dimensional spectral analysis of the phase fraction at $t = 1.72\,\mathrm{s}$ from (top) unbounded IPCS, (middle) bounded IPCS and (bottom) `twoPhaseEulerFoam` solvers. Left: histogram of power spectral density. Right: radially-averaged power spectrum. 61

5.6 Surface plot of (left) phase fraction, (center) gas velocity and (right) liquid velocity at $t = 1.72\,\mathrm{s}$ from bounded IPCS with interfacial pressure. 62

5.7 Two-dimensional spectral analysis of the phase fraction at $t = 1.72\,\mathrm{s}$ from bounded IPCS with interfacial pressure. Left: histogram of power spectral density. Right: radially-averaged power spectrum. 63

5.8 Time-evolution of gas holdup. 64

6.1 Simulation domain for gas-liquid flow inside a channel with the diffuse-interface method. 72

6.2 Simulation domain for gas-liquid flow past a stationary cylinder with the diffuse-interface method. 73

6.3 Surface plot of α_g at $t = 1.72\,\mathrm{s}$ with hyperbolic tangent diffuse-interface and $\epsilon = 0.02$. The grayscale colorbar denotes the phase-field that describes the diffuse-interface, thresholded to show $\phi \geq -0.999$. 76

6.4 Surface plot of (left) α_g, (center) gas velocity and (right) liquid velocity at $t = 1.72\,\mathrm{s}$ with hyperbolic tangent diffuse-interface and $\epsilon = 0.02$. 77

6.5 Comparison of diffuse-interface width generated using the same hyperbolic tangent function with varying ϵ. 78

6.6 Surface plot of α_g at $t = 1.72\,\mathrm{s}$ with hyperbolic tangent diffuse-interface and $\epsilon = 0.01$. The grayscale colorbar denotes the phase-field that describes the diffuse-interface, thresholded to show $\phi \geq -0.999$. 79

6.7 Surface plot of α_g at $t = 1.72\,\mathrm{s}$ with hyperbolic tangent diffuse-interface and $\epsilon = 0.1$. The grayscale colorbar denotes the phase-field that describes the diffuse-interface, thresholded to show $\phi \geq -0.999$. 80

6.8 Surface plot of (left) α_g, (center) gas velocity and (right) liquid velocity at $t = 1.72\,\mathrm{s}$ with hyperbolic tangent diffuse-interface and $\epsilon = 0.01$. 81

6.9 Surface plot of (left) α_g, (center) gas velocity and (right) liquid velocity at $t = 1.72\,\mathrm{s}$ with hyperbolic tangent diffuse-interface and $\epsilon = 0.1$. 81

6.10 Time evolution of overall gas hold-up inside a channel with solid boundaries defined by a hyperbolic tangent diffuse-interface. 82

6.11 α_g profile along $y = 0.08\,\mathrm{m}$ with different hyperbolic tangent diffuse-interface widths. 83

6.12 Comparison of diffuse-interface generated using hyperbolic tangent and piece-wise cosine functions with $\epsilon = 0.02$ and $\eta = \epsilon \tanh^{-1}(0.999)$. 85

6.13 Surface plot of (left) phase fraction, (center) gas velocity and (right) liquid velocity at $t = 1.72\,\mathrm{s}$ with piece-wise cosine diffuse-interface and (top) $\epsilon = 0.01$, (middle) $\epsilon = 0.02$ and (bottom) $\epsilon = 0.1$. 86

6.14 Time evolution of overall gas hold-up inside a channel with solid boundaries defined by a piece-wise cosine diffuse-interface. 87

6.15 α_g profile along $y = 0.08\,\mathrm{m}$ with different piece-wise cosine diffuse-interface widths. 88

6.16 Error in α_g profile along $y = 0.08\,\mathrm{m}$ as a function of ϵ. 89

6.17 Evolution of gas-liquid flow past a stationary cylinder with a boundary-conformal mesh. Streamlines are of (top) gas and (bottom) liquid phases. . 91

6.18 Evolution of gas-liquid flow past a stationary cylinder with a hyperbolic tangent diffuse-interface and $\epsilon = 0.01$. The diffuse-interface is in grayscale and streamlines are of (top) gas and (bottom) liquid velocities. 93

6.19 Evolution of gas-liquid flow past a stationary cylinder with a piece-wise cosine diffuse-interface and $\epsilon = 0.01$. The diffuse-interface is in grayscale and streamlines are of (top) gas and (bottom) liquid velocities. 94

6.20 Time evolution of overall gas hold-up in flow past a stationary cylinder. . . 95

A.1 Time evolution of gas hold-up for different number of mesh elements. . . . 118

A.2 Time evolution of gas hold-up obtained from the hyperbolic tangent diffuse-interface simulations of channel flow using different number of mesh elements. 118

List of Symbols

α	Time-averaged local phase fraction
Δt	Time step
δ	Artificial compressibility
ϵ	Diffuse-interface width parameter
ϵ_l	Local error tolerance
η	Diffuse-interface width
$\frac{D_i \boldsymbol{.}}{Dt}$	Material derivative, $\frac{\partial \boldsymbol{.}}{\partial t} + \boldsymbol{v}_i \cdot \boldsymbol{\nabla .}$
Γ	Simulation domain boundary
$\langle \boldsymbol{.}, \boldsymbol{.} \rangle$	Inner product
$\langle \boldsymbol{.} \rangle$	Volume average
μ	Dynamic viscosity
ν	Kinematic viscosity
Ω	Simulation domain
$\bar{\boldsymbol{.}}$	Time average (Reynolds average)
$\bar{\bar{\boldsymbol{.}}}$	Phasic average, $\bar{\boldsymbol{.}}/\alpha_i$

ϕ	Phase-field
ρ	Density
$\tilde{\boldsymbol{x}}_c$	Scaled diffuse-interface position vector
φ	Trial function
$\boldsymbol{\tau}$	Stress tensor
$\boldsymbol{g}$	Gravitational acceleration
$\boldsymbol{M}$	Momentum source term
$\boldsymbol{v}$	Velocity vector
$\boldsymbol{x}$	Position vector
$\widehat{\blacksquare}$	Mass-weighted mean phasic average (Favre average), $\overline{\rho_i \blacksquare}/\overline{\rho_i}$
a_j	Backward-differentiation coefficient
b_j	Adams-Bashforth coefficient
C_D	Drag coefficient
C_L	Lift coefficient
C_W	Wall coefficient
C_P	Interfacial pressure coefficient
C_{VM}	Virtual mass coefficient
d	Diameter
Eu	Euler number
Fr	Froude number

h	Mesh element length
P	Pressure
Pe	Péclet number
r_d	Volume to projected area ratio
Re	Reynolds number
t	Time
w	Test function
y_a	Trial solution

Chapter 1

Introduction

1.1 Research Motivation

Two-phase gas-liquid flow systems are prevalent in industrial processes and, consequently, the hydrodynamical behavior of gas-liquid systems is of significant interest. Examples of gas-liquid flow systems include bubble columns [1–4], loop reactors [5, 6], nuclear reactor cores [7, 8], cyclones [9], hydrocarbon pipelines [10–12] and disengagers [13]. The hydrodynamical behavior of gas-liquid systems is complicated and dependent on many factors [14, 15], which are both difficult to control and measure experimentally. Thus, the use of experimentation alone to study gas-liquid hydrodynamics is costly, time-consuming and does not provide access to the dynamic spatially varying flow field. Computational fluid dynamics (CFD) is a necessary approach to augmenting experimental research in that it addresses these challenges.

Significant progress has been made in the development of models of gas-liquid flows and their application to CFD simulations of multiphase flows [16–20]. Information on important process information such as pressure drop, separation efficiency, temperature distribution and reaction yield can be obtained from multiphase CFD simulations. The simulations can be used to study process sensitivity to the operating conditions as well as a screening tool for researchers to devise experimental set-ups involving conditions that merit

further study. In addition to studying the hydrodynamical behavior of gas-liquid systems, CFD can also be used to design equipment through traditional iterative methods or more advanced automated topological optimization methods [21–27] by performing simulations of different designs.

There are two important aspects to performing CFD simulations of multiphase flows: the choice of the multiphase modeling approach and the specification of the geometry. The choice of multiphase modeling approach will depend on the system of interest as certain methods are better-suited for specific systems. Factors such as computational cost and the desired information are crucial when choosing the appropriate method. The performance of the CFD simulations is also contingent on proper specification of the domain geometry. In industrial processes, the domain geometry will correspond to that of the equipment of interest, which are often highly complex. How this geometry is specified is essential to the performance and stability of the simulations.

Current approaches to modeling two-phase flows fall into three main categories: two-fluid (Euler-Euler), Euler-Lagrange and interface-tracking models. Of the three family of methods, the two-fluid model is the least computationally intensive and is thus an attractive approach to modeling gas-liquid systems. The two-fluid model approximates the fluids as inter-penetrating continua with conservation equations formulated for each phase [16]. This method requires interphase momentum transfer constitutive relationships that describe the momentum exchange between each phase. The selection of these constitutive relationships has been shown to affect the simulation results significantly [28, 29].

In the Euler-Lagrange approach, the fluid is treated as a continuum but the dispersed phase is treated as discrete particles whose motions are determined through simultaneously solving Newton's equations of motion for each particle [1, 30–33]. This has the benefit of accurately resolving the motion of the individual particles/droplets/bubbles. However, this benefit has the added cost of the computational requirement for computing the equations of motion for the dispersed phase being significantly higher than that of the two-fluid model [32].

Interface-tracking multiphase models include volume-of-fluid (VOF) [34], marker-and-cell (MAC), level-set [35] and phase-field methods [36–43]. These methods are well suited

for systems with segregated flows such as systems involving slug flow [44, 45]. However, for dispersed systems, these approaches require the resolution of all of the interfaces between the fluid and the dispersed phase, which is typically infeasible for industrially relevant scales [46, 47].

A crucial aspect in the use of CFD for design and optimization of process equipment is the specification of internal physical features, which can have highly complex shapes. These features need to be specified as physical boundaries in the simulation, which can be achieved by either using a conformal mesh or an embedded domain method. With a conformal mesh, the geometry is defined such that once generated, the mesh surfaces correspond to the physical boundaries. This process can be tedious, time-consuming and is computationally challenging and subject to numerical instability for complicated geometries. Additionally, if the internal features are changed, which is likely the case during the design and optimization process, the mesh will also have to change, thus requiring the mesh to be regenerated. In the case of moving mesh problems, methods like the arbitrary Lagrangian-Eulerian (ALE) method [48] are used, but ALE requires the mesh to be deformed as the boundary moves.

Instead of using a conformal mesh, the physical boundaries can be "embedded" in the problem using methods such as fictitious domain [49], immersed boundary [50, 51] and diffuse domain/interface [52–56] methods. The physical boundaries are defined in the embedded domain method through the use of a level-set function, a phase-field, *etc.* Since the physical boundaries are not explicitly defined by the domain mesh, it is not required to conform to the physical boundaries and a simple structured mesh can be used. This has the benefit of avoiding skewed elements that are detrimental to the numerical accuracy of the solution. The ease in which the internal features can be modified during the simulation is highly beneficial when optimizing a design since the field that describes the boundaries can directly be modified by the optimization scheme.

Physical boundaries that are defined using fictitious domain and immersed boundary methods are generally sharp boundaries whose effect may be approximated through distribution of the boundary over a number of mesh elements. This requires the solution field to be interpolated from the physical boundary to the nearest neighboring node/cell [57]. Special consideration must also be paid when handling mesh elements that are cut by

the embedded boundaries [56]. The diffuse domain/interface method defines the physical boundaries using a phase-field. For example, the phase-field can vary between zero and one [56]:

$$\phi = \begin{cases} 1, & \text{physical domain,} \\ 0, & \text{otherwise,} \end{cases} \tag{1.1}$$

where ϕ is the phase-field.

As previously discussed, the phase-field approach where the phase-field describes the gas-liquid interface in a thermodynamically consistent manner has been extensively used to model multiphase flows [36–43]. This method does not require for the solution to be interpolated at every time step since the phase-field ensures the smooth transition from the fluid to the solid phase. The boundary normal vector is computed from the gradient of the phase-field, aiding in the imposition of Neumann boundary conditions. The diffuse-interface method is also agnostic to the numerical method used to solve the governing equations. Based on these factors, the diffuse-interface method is an attractive approach in imposing physical boundaries in simulations involving multiphase flow modeled using the two-fluid model.

1.2 Objectives

The overall objective of this research is to develop, verify and validate a diffuse-interface method for imposing physical boundaries when solving the two-fluid model equations.

To complete the aforementioned objectives, the following studies have been conducted:

1. Development, verification and partial validation of a phase fraction-bounded finite element method for the solution of the two-fluid model equations using modified incremental pressure correction scheme (TFM-IPCS).

2. Development, verification and partial validation of a diffuse-interface blended method for the imposition of physical boundaries.

1.3 Structure of Thesis

This thesis is organized into seven chapters: Chapter 2 – background, Chapter 3 – literature review, Chapter 4 – finite element formulation for the solution of the two-fluid model, Chapter 5 – phase-bounded finite element method for two-fluid incompressible flow systems, Chapter 6 – diffuse-interface blended method for physical boundaries and Chapter 7 – conclusions and recommendations for future work.

Chapter 2 describes the relevant theoretical background to the studies. The two-fluid model is introduced along with the background on the numerical method used in the studies. Solution methods for single-phase incompressible Navier-Stokes equations are also described.

Chapter 3 provides an overview of the current literature on the different solution methods for the two-fluid model and embedded domain methods for multiphase flow systems. Methods to address the numerical complexities of the two-fluid model such as phase fraction boundedness and the ill-posedness of the model are also discussed.

Chapter 4 presents the finite element formulation for the solution of the two-fluid model that is used in the studies presented in Chapters 5 and 6. This includes the numerical scheme used for the time discretization and the weak formulation of the governing equations.

The solution method presented in Chapter 4 is then used in Chapter 5 in conjunction with a novel approach to ensuring phase fraction boundedness in the two-fluid model. Simulation results of a gas-liquid bubble column with and without the boundedness constraint are presented and compared to results from the two-fluid solver from the package `OpenFOAM`. The effect of the common assumption that the two phases share the same pressure field even in bubbly flow is also included in this chapter.

Chapter 6 presents a diffuse-interface approach for imposing physical boundary conditions. The governing equations are solved using the solution method presented in Chapter 5. Results from two-phase simulations of a bubble column and flow past a cylinder are presented. Validation of the diffuse-interface method using boundary-conformal mesh simulations is also presented in this chapter.

Lastly, Chapter 7 summarizes the conclusions from this work and recommendations for future work.

Chapter 2

Background

2.1 Gas-Liquid Hydrodynamics

The multiphase system of interest in this work is the gas-liquid flow system. Different flow regimes exist for gas-liquid flow systems that are dependent on the configuration of the system in question. The main gas-liquid systems that are of industrial interest include flow in horizontal pipes and vertical pipe flow [58]. This section will provide an overview of the different flow regimes in the aforementioned gas-liquid flow systems.

2.1.1 Horizontal Flow

Horizontal gas-liquid flow is commonly found in oil and gas pipelines and power plants [10]. The different flow regimes in horizontal pipe flow are shown in Fig. 2.1. At low gas and liquid velocities, stratified flow is present [59]. In stratified flow, only the liquid phase is present at the bottom of the pipe and the gas-liquid interface has minimal curvature. As the gas velocity increases, waves such as Kelvin-Helmholtz waves appear [58] (wave flow region in Fig. 2.1). If the gas velocity is further increased, atomization will occur and a turbulent film can form, resulting in annular flow [58]. When the liquid velocity increases, the waves can form a liquid bridge as it touches the top of the pipe, resulting in liquid

slugs forming. Slugs can cause vibrations in pipes and are undesirable in a piping system [10]. At high liquid velocities but low gas velocities, bubble flow can occur and the gas bubbles will be concentrated near the top of the pipe due to buoyancy force [58].

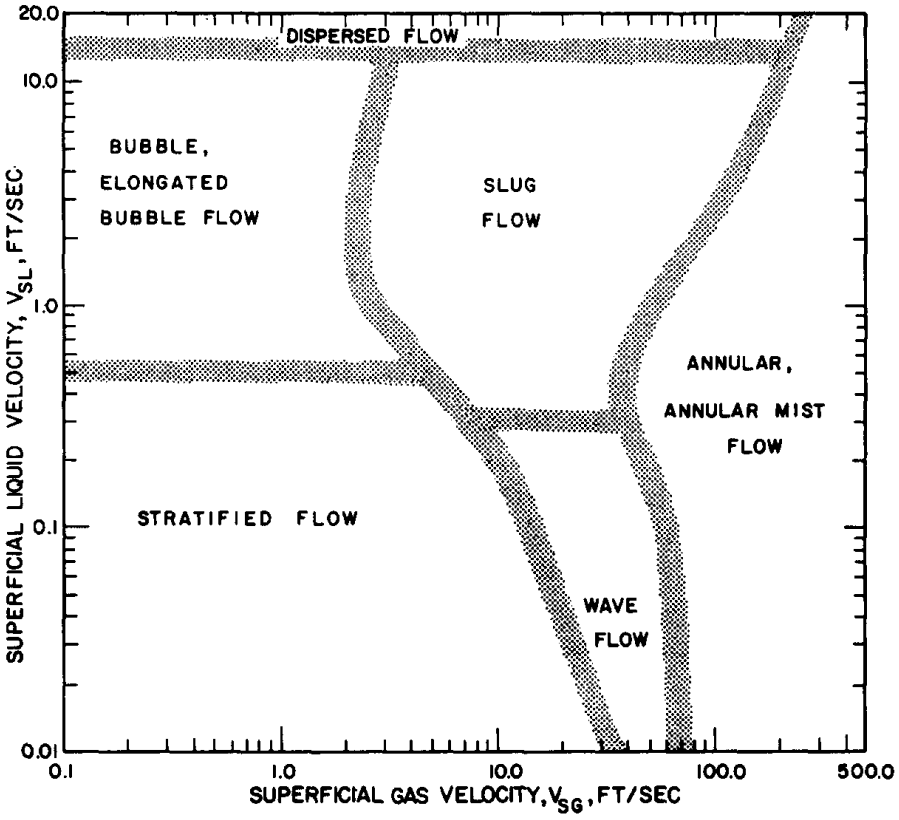


Figure 2.1: Different flow regimes in gas-liquid flow in horizontal pipes [60].

2.1.2 Vertical Flow

Vertical gas-liquid flow can be classified into two main systems: bubble columns (superficial $\boldsymbol{v}_l = 0$) and gas-liquid cocurrent flow (superficial $\boldsymbol{v}_l \neq 0$). Bubble columns are readily found in chemical engineering applications [1] while gas-liquid cocurrent flow is found in nuclear reactors [61]. Figure 2.2 shows a flow regime map for gas-liquid bubble columns as a

function of the superficial gas velocity and pipe diameter. The flow regime as a function of liquid and gas superficial velocities for cocurrent upward flow is shown in Fig. 2.3.

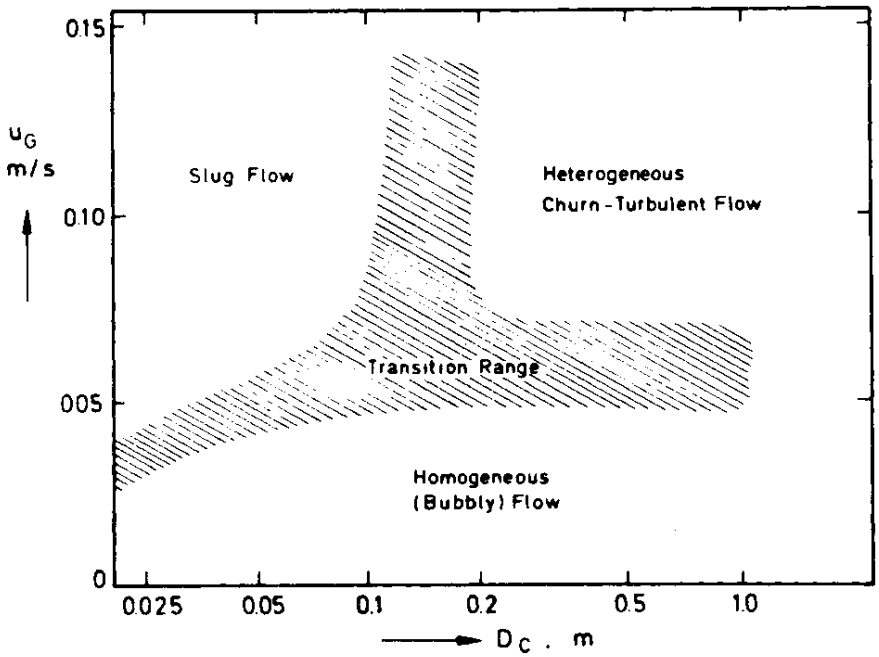


Figure 2.2: Flow regime as a function of gas superficial velocity and column diameter [15].

Gas-liquid flows in vertical pipes can be classified into the following flow regimes [14, 58, 62, 63]: dispersed bubble, vortical-spiral (transitional) flow, turbulent (also referred to as churn-turbulent or slug) and annular flow regimes. In the dispersed bubble regime, the bubble size is relatively uniform due to very little coalescence occurring in the column. This type of multiphase flow corresponds to low gas velocities, where the liquid is carried upwards near the bubbles and falls downward in the area between the bubbles (Fig. 2.4 left). As the gas velocity increases, the bubbles start to move in clusters, exhibiting collective behavior, which eventually leads to coalesce into large segregated regions. The larger bubbles move in a spiral manner while the smaller bubbles move up and down near the wall (Fig. 2.4 center). This regime is the transition between dispersed flow to fully turbulent flow (Fig. 2.4 right). In turbulent flow, the bubble coalescence is increased and results in large slugs that disrupt the continuous nature of the flow. In smaller pipes, slug

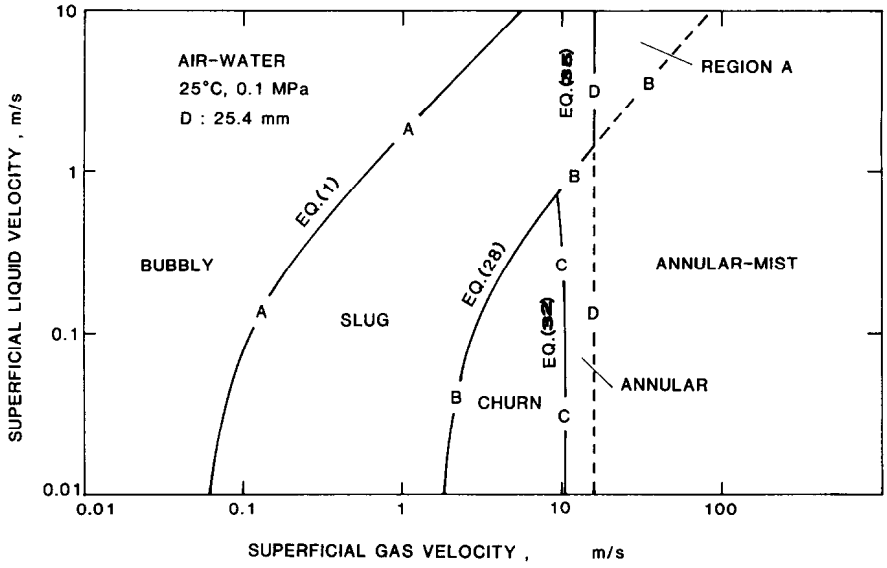


Figure 2.3: Different flow regimes in cocurrent gas-liquid flow in vertical pipes [61].

flow is present and in larger pipes, a heterogeneous phase such as the one shown in Fig. 2.4 is present [15]. Annular flow occurs when the gas velocity is further increased and the slugs coalesce [58].

2.2 Two-Fluid Model

In modeling gas-liquid flows using the two-fluid model, each of the phases in the system is considered to be a continuous fluid. Each of the phases has its own set of conservation equations that are coupled together through interphase transfer terms. It is impractical to solve for the local instantaneous motion of the fluid, thus averaging schemes are used to solve for the macroscopic flow behavior instead [16]. Time-averaged quantities are denoted by an overbar, $\bar{\cdot}$. A double-overbar, $\bar{\bar{\cdot}}$, denotes a phasic average quantity, which are defined as the time-averaged quantity divided by the phase fraction, $\bar{\cdot}/\alpha_i$. A hat, $\widehat{\cdot}$, denotes a mass-weighted mean phasic average quantity (Favre average), defined as $\overline{\rho_i \cdot}/\overline{\rho_i}$.

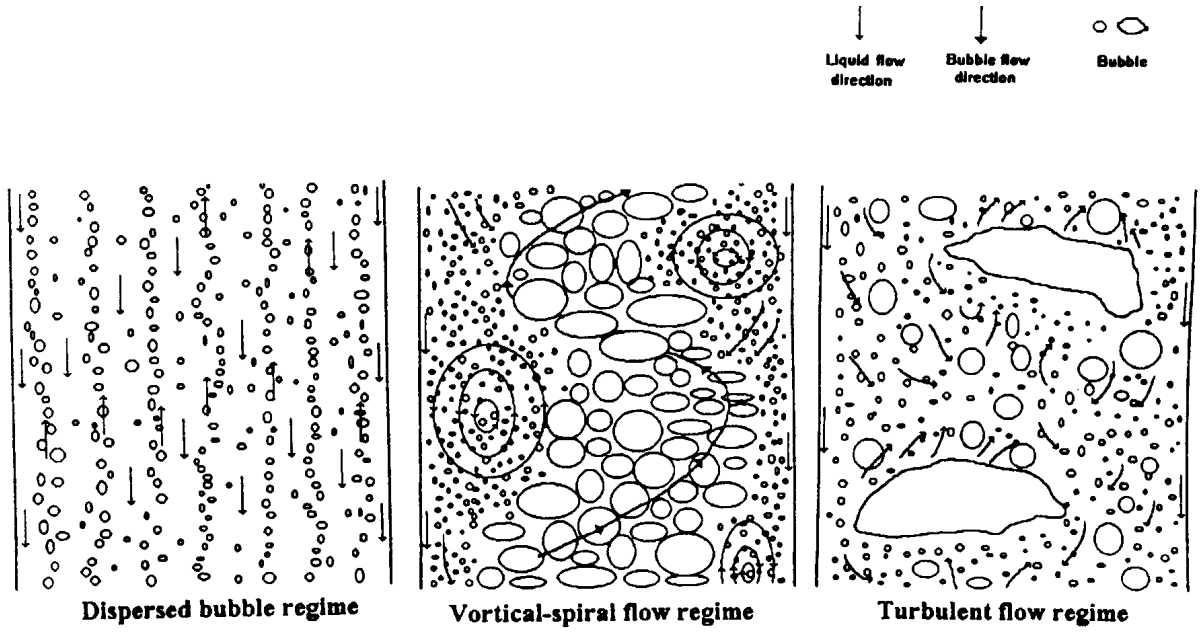


Figure 2.4: Different flow regimes in gas-liquid bubble columns [14].

2.2.1 Mass Conservation

The general expression for the conservation of mass for a phase q, in the absence of inter-phase mass transfer and reaction, is given as follows [16]:

$$\frac{\partial(\alpha_q \overline{\overline{\rho_q}})}{\partial t} + \boldsymbol{\nabla} \cdot \left(\alpha_q \overline{\overline{\rho_q}} \widehat{\boldsymbol{v}}_q\right) = 0, \tag{2.1}$$

where α_q is the time-averaged local phase fraction of phase q, $\overline{\overline{\rho_q}}$ is the time-averaged phasic average density and $\widehat{\boldsymbol{v}}_q$ is the time-averaged mass-weighted mean phase velocity.

2.2.2 Momentum Conservation

The conservation of momentum for phase q is given as [16]:

$$\frac{\partial\left(\alpha_q\overline{\overline{\rho_q}}\widehat{\boldsymbol{v}}_q\right)}{\partial t}+\boldsymbol{\nabla}\cdot\left(\alpha_q\overline{\overline{\rho_q}}\widehat{\boldsymbol{v}}_q\widehat{\boldsymbol{v}}_q\right)=-\boldsymbol{\nabla}\left(\alpha_q\overline{\overline{P_q}}\right)+\boldsymbol{\nabla}\cdot\left(\alpha_q\overline{\overline{\boldsymbol{\tau}_q}}\right)+\alpha_q\overline{\overline{\rho_q}}\widehat{\boldsymbol{g}}_q+\boldsymbol{M}_q+\overline{\overline{P_{q,i}}}\boldsymbol{\nabla}\alpha_q-\boldsymbol{\nabla}\alpha_q\cdot\overline{\overline{\boldsymbol{\tau}_{q,i}}},\tag{2.2}$$

where $\overline{\overline{P_q}}$ is the time-averaged phasic pressure, $\overline{\overline{\boldsymbol{\tau}_q}}$ is the time-averaged phasic viscous stress tensor, $\widehat{\boldsymbol{g}_q}$ is the time-averaged mass-weighted mean phase gravitational acceleration, $\boldsymbol{M}_q$ is the interphase momentum source term, $\overline{\overline{P_{q,i}}}\boldsymbol{\nabla}\alpha_q$ and $\boldsymbol{\nabla}\alpha_q\cdot\overline{\overline{\boldsymbol{\tau}_{q,i}}}$ are the contributions of interfacial stresses and the subscripts indicate the fluid-dispersed phase pairing and type of force, respectively. In the dispersed flow regime, the interfacial pressure and shear stress of the continuous, c, and dispersed, d, phases can be assumed to be equal to each other $\overline{\overline{P_{c,i}}}\approx\overline{\overline{P_{d,i}}}=\overline{\overline{P_{int}}}$ (neglecting surface tension effects) and $\overline{\overline{\boldsymbol{\tau}_{c,i}}}\approx\overline{\overline{\boldsymbol{\tau}_{d,i}}}$ [16, 64]. Additionally, the pressure of the dispersed phase can be approximated by the interfacial pressure, $\overline{\overline{P_d}}\approx\overline{\overline{P_{int}}}$ [16]:

$$\frac{\partial\left(\alpha_c\overline{\overline{\rho_c}}\widehat{\boldsymbol{v}}_c\right)}{\partial t}+\boldsymbol{\nabla}\cdot\left(\alpha_c\overline{\overline{\rho_c}}\widehat{\boldsymbol{v}}_c\widehat{\boldsymbol{v}}_c\right)=-\alpha_c\boldsymbol{\nabla}\overline{\overline{P_c}}+\boldsymbol{\nabla}\cdot\left(\alpha_c\overline{\overline{\boldsymbol{\tau}_c}}\right)+\alpha_c\overline{\overline{\rho_c}}\widehat{\boldsymbol{g}}+\boldsymbol{M}_c+\left(\overline{\overline{P_{int}}}-\overline{\overline{P_c}}\right)\boldsymbol{\nabla}\alpha_c-\boldsymbol{\nabla}\alpha_q\cdot\overline{\overline{\boldsymbol{\tau}_{c,i}}},\tag{2.3a}$$

$$\frac{\partial\left(\alpha_d\overline{\overline{\rho_d}}\widehat{\boldsymbol{v}}_d\right)}{\partial t}+\boldsymbol{\nabla}\cdot\left(\alpha_d\overline{\overline{\rho_d}}\widehat{\boldsymbol{v}}_d\widehat{\boldsymbol{v}}_d\right)=-\alpha_d\boldsymbol{\nabla}\overline{\overline{P_{int}}}+\boldsymbol{\nabla}\cdot\left(\alpha_d\overline{\overline{\boldsymbol{\tau}_d}}\right)+\alpha_d\overline{\overline{\rho_d}}\widehat{\boldsymbol{g}}+\boldsymbol{M}_d-\boldsymbol{\nabla}\alpha_d\cdot\overline{\overline{\boldsymbol{\tau}_{c,i}}}.\tag{2.3b}$$

The effect of the interfacial shear stress is significant in the segregated flow regime [65]. In this work, the focus will be on the dispersed flow regime, therefore, the interfacial shear stress contribution is assumed to be negligible. The conservation of momentum for a dispersed flow system is thus:

$$\frac{\partial\left(\alpha_c\overline{\overline{\rho_c}}\widehat{\boldsymbol{v}}_c\right)}{\partial t}+\boldsymbol{\nabla}\cdot\left(\alpha_c\overline{\overline{\rho_c}}\widehat{\boldsymbol{v}}_c\widehat{\boldsymbol{v}}_c\right)=-\alpha_c\boldsymbol{\nabla}\overline{\overline{P_c}}+\boldsymbol{\nabla}\cdot\left(\alpha_c\overline{\overline{\boldsymbol{\tau}_c}}\right)+\alpha_c\overline{\overline{\rho_c}}\widehat{\boldsymbol{g}}+\boldsymbol{M}_c+\left(\overline{\overline{P_{int}}}-\overline{\overline{P_c}}\right)\boldsymbol{\nabla}\alpha_c, \tag{2.4a}$$

$$\frac{\partial\left(\alpha_d\overline{\overline{\rho_d}}\widehat{\boldsymbol{v}}_d\right)}{\partial t}+\boldsymbol{\nabla}\cdot\left(\alpha_d\overline{\overline{\rho_d}}\widehat{\boldsymbol{v}}_d\widehat{\boldsymbol{v}}_d\right)=-\alpha_d\boldsymbol{\nabla}\overline{\overline{P_{int}}}+\boldsymbol{\nabla}\cdot\left(\alpha_d\overline{\overline{\boldsymbol{\tau}_d}}\right)+\alpha_d\overline{\overline{\rho_d}}\widehat{\boldsymbol{g}}+\boldsymbol{M}_d. \tag{2.4b}$$

For the sake of brevity, the average notations are omitted in the subsequent sections. The quantities are assumed to be averaged quantities corresponding to the definition above.

2.2.3 Interfacial Forces

The interphase momentum transfer term $\boldsymbol{M}_q$, defined as the transfer of momentum *into* phase q, is the sum of the contributions from the different modes of momentum transfer: drag, lift, virtual mass, wall lubrication, *etc.* The momentum transfer term for the continuous phase with the aforementioned contributions is [16, 66, 67]:

$$\boldsymbol{M}_c=\boldsymbol{M}_{c,drag}+\boldsymbol{M}_{c,lift}+\boldsymbol{M}_{c,virtual\ mass}+\boldsymbol{M}_{c,wall}+\boldsymbol{M}_{c,other}. \tag{2.5}$$

Given that the momentum exchange between the phases should sum to zero, momentum transfer of the dispersed phase d is given as:

$$\boldsymbol{M}_c=-\boldsymbol{M}_d. \tag{2.6}$$

Drag Force

The drag force term in Eqn. (2.5) is the sum of the form and skin drag forces which are due to the imbalance of pressure and shear forces at the interface, respectively [16]. Drag acts in the opposite direction of the relative motion of the bubble/particle. The interphase momentum transfer of phase c due to drag for a dispersed phase d in fluid c, $\boldsymbol{M}_{c,drag}$, is

given as [16]:

$$\boldsymbol{M}_{c,drag} = \frac{1}{2}\rho_c \alpha_d \frac{C_D}{r_d} \|\boldsymbol{v}_r\| \boldsymbol{v}_r, \tag{2.7}$$

where r_d is the ratio of the volume to projected area of the bubble/particle, C_D is the drag coefficient and $\boldsymbol{v}_r$ is the relative velocity between the dispersed and continuous phases, $\boldsymbol{v}_r = \boldsymbol{v}_d - \boldsymbol{v}_c$. For spherical bubbles/particles:

$$\boldsymbol{M}_{c,drag} = \frac{3}{4}\rho_c \alpha_d \frac{C_D}{d_d} \|\boldsymbol{v}_r\| \boldsymbol{v}_r, \tag{2.8}$$

where d_d is the bubble/particle diameter.

Lift Force

Lift is the force exerted on the bubbles/particles that governs the transverse movement of the dispersed phase in a fluid and is a result of shear forces and the asymmetric pressure distribution around the dispersed particle/bubble [28, 68, 69]. The direction of lift is perpendicular to the direction of flow. The expression for the momentum transfer to the continuous fluid c due to lift is given as [69, 70]:

$$\boldsymbol{M}_{c,lift} = C_L \rho_c \alpha_d \boldsymbol{v}_r \times (\boldsymbol{\nabla} \times \boldsymbol{v}_c), \tag{2.9}$$

where C_L is the lift coefficient.

Virtual Mass Force

The virtual mass (or added mass) force is related to the acceleration of one phase in relation to another [69]. As the dispersed phase is accelerating in the continuous phase, it is displacing the surrounding fluid, increasing the inertia. The momentum transfer to phase c due to virtual mass force is given as [69, 70]:

$$\boldsymbol{M}_{c,virtual\ mass} = \alpha_d \rho_c C_{VM} \left(\frac{D_d \boldsymbol{v}_d}{Dt} - \frac{D_c \boldsymbol{v}_c}{Dt} \right), \tag{2.10}$$

where C_{VM} is the virtual mass coefficient, D_d/Dt and D_c/Dt are the material derivatives with respect to phases d and c, respectively.

Wall Lubrication Force

The wall lubrication force is a wall effect that occurs in bubbly flow where the continuous phase wets the walls. It occurs when a bubble's proximity to the wall results in asymmetric drainage of fluid around the bubble. The side that is close to the wall will drain slower due to the no-slip condition. The asymmetry creates a hydrodynamic force normal to the wall that pushes the bubble away from the wall [67]. The wall lubrication force is given as [7]:

$$\boldsymbol{M}_{c,wall} = -C_W \alpha_d \rho_c \|\boldsymbol{v}_r - (\boldsymbol{v}_r \cdot \boldsymbol{n}_W)\boldsymbol{n}_W\|^2 \boldsymbol{n}_W, \tag{2.11}$$

where C_W is the wall coefficient and $\boldsymbol{n}_W$ is the unit normal outward on the wall.

Interfacial Pressure

The interfacial pressure is determined from a volume average of the solution of potential flow around a single sphere [67, 71]. This interfacial pressure is given by:

$$P_{c,i} = P_c - C_P \rho_c \boldsymbol{v}_r \cdot \boldsymbol{v}_r, \tag{2.12}$$

where C_P is the interfacial pressure coefficient representing the flow near the bubble and the shape of the bubble [71]. For the case where the particle/bubble size distribution is uniform, the inviscid flow solution is used to approximate the value of the interfacial pressure coefficient, $C_P = 0.25$ [64, 72].

2.3 Numerical Method for Partial Differential Equations

The governing equations in the two-fluid model are partial differential equations (PDEs). Generally, analytical solutions to PDEs only exist for unique, simplified cases. Thus, the solution to the PDEs in the two-fluid model is obtained numerically. In this section, the numerical method for the solution of partial differential equations is introduced along with some common solution methods for single-phase incompressible Navier-Stokes equations.

2.3.1 Method of Weighted Residuals

The method of weighted residuals is a method to approximate solutions for ordinary and partial differential equations. The solution to the differential equation is approximated by a parameterized approximate solution, the *trial solution*. The trial solution to the equation is expressed in terms of trial or basis functions, for example [73]:

$$y_a(\boldsymbol{x}) = \sum_{i=1}^{N} a_i \varphi_i(\boldsymbol{x}), \tag{2.13}$$

where y_a is the trial solution, a_i is a set of coefficients that will help the trial solution satisfy the differential equation, φ_i is the trial (basis) function that satisfies the boundary conditions and N is the number of trial functions. The residual is defined as [73]:

$$L(y_a) = R, \tag{2.14}$$

where the left-hand side represents the differential equation where if y is the exact solution to the differential equation, $L(y) = 0$, and R is the residual. Substituting the trial function in Eqn. (2.13) will give a residual error function:

$$R(\boldsymbol{x}) = L\left(\sum_{i=1}^{N} a_i \varphi_i(\boldsymbol{x})\right). \tag{2.15}$$

From Eqn. (2.15), the only parameters on the right-hand side that can be changed for a given trial function are $\{a_i\}$, the determination of which results in an approximation to the solution of the differential equation.

When the residual error function is zero over its domain the solution is exact, thus any numerical solution should have a residual that obeys this constraint within acceptable tolerances. The residual error over the entire solution domain needs to be minimized, but the residual defined in Eqn. (2.15) is a function of position. To minimize the residual over the entire domain, the residual needs to be integrated. This is achieved by computing the weighted integral over the domain [74]:

$$\langle R(\boldsymbol{x}), w_j(\boldsymbol{x})\rangle_\Omega = \int_\Omega R(\boldsymbol{x}) w_j(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} = 0, \tag{2.16}$$

where $\langle \boldsymbol{.}, \boldsymbol{.} \rangle$ is the inner product operator, w_j is the weighting (test) function and Ω represents the domain. The choice of weighting function can vary and result in different numerical methods belonging to the family of the method of weighted residuals such as, subdomain, least squares, Galerkin, moment and collocation methods [73]. Finite volume and finite element are examples of subdomain and Galerkin methods, respectively, and are commonly used in solving fluid flow problems.

2.3.2 Finite Element Method

In this work, the governing equations are solved using the Galerkin finite element method (FEM). The governing equations are reformulated into their weak formulations, where the condition that the solution to a differential equation must satisfy the differential equation at every point in the domain is relaxed. Instead, the solution will weakly (i.e. on average over the whole domain) satisfy the differential equation. If the solution to the differential equation is smooth, the solution to the differential equation is also the solution of the weak formulation [75]. The weak formulation is obtained through the method of weighted residuals and takes the form of Eqn. (2.16). For the case of the Galerkin method, the test

function is from the same family of functions as the basis function(s) [73, 75]:

$$w_j = \varphi_j(\boldsymbol{x}). \tag{2.17}$$

Using the following Poisson problem as an example:

$$-\nabla^2 y = f \quad \text{on } \Omega, \qquad \boldsymbol{n} \cdot \boldsymbol{\nabla} y = h \quad \text{on } \Gamma_N, \tag{2.18}$$

where Ω is the domain and Γ is the boundary of the domain. Taking the inner product of the residual function with the test function and using integration by parts, the weak formulation of the problem is as follows:

$$\int_\Omega \boldsymbol{\nabla}\varphi \cdot \boldsymbol{\nabla} y \mathrm{d}\Omega - \int_\Omega \varphi f \mathrm{d}\Omega - \int_{\Gamma_N} \varphi h \mathrm{d}\Gamma = 0. \tag{2.19}$$

The choice of the basis functions can affect the accuracy and convergence of the solution [75]. In FEM, the basis functions are local interpolating functions over parts of the domain. These subdomains are known as "elements" and the solution is represented in terms of piece-wise interpolating functions, which often are low order polynomials. An example of this representation with linear (first order polynomial) basis functions for an arbitrary function y_a with the same form as that of Eqn. (2.13) is given in Fig. 2.5. The basis functions at nodes 1–3 are piece-wise linear functions whose values are one at the node and zero elsewhere. The coefficient a_i in Eqn. (2.13) corresponds to the value of the function at that particular node. From Fig. 2.5, it is clear that the accuracy of the representation increases as the spacing between the nodes decreases.

In FEM jargon, the piece-wise polynomials that make up each element are called element shape functions. The element shape functions are defined on a reference coordinate system, (ξ, η) for two-dimensional problems (Fig. 2.6). Each mesh element can be mapped to the reference coordinate and the corresponding linear system to the differential equation can be constructed in terms of the reference element [76]:

$$\boldsymbol{A}^e \boldsymbol{K}^e \boldsymbol{y} = \boldsymbol{A}^e \boldsymbol{f}^e, \tag{2.20}$$

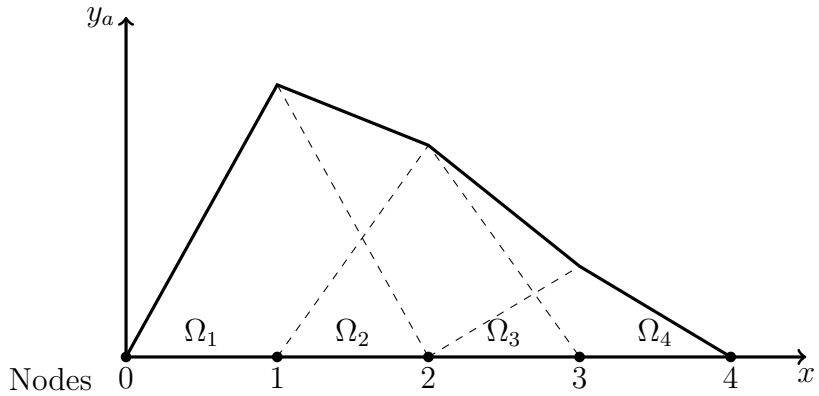


Figure 2.5: Linear piece-wise representation of y_a (solid line) and the corresponding basis functions (dashed lines). Figure adapted from Ref. [75].

where $\boldsymbol{A}^e$ is an assembly operator that is applied to the local element matrix, $\boldsymbol{K}^e$, and nodal vector, $\boldsymbol{f}^e$, and $\boldsymbol{y}$ is a vector of unknowns. $\boldsymbol{K}^e$ and $\boldsymbol{f}^e$ are quantities that correspond to the linear system if the domain is the reference element. The assembly operator maps $\boldsymbol{K}^e$ and $\boldsymbol{f}^e$ from the reference (local) element to the actual (global) mesh element [76].

Compared to the other popular spatial discretization methods such as finite difference and finite volume, the finite element method has several benefits. FEM is a more general method that can handle unstructured meshes with much more ease than the other two methods. Unlike the finite volume method that is locally conservative over each subdomain, the finite element method is globally conservative over the entire domain and the residual is globally satisfied. Increasing the order of the interpolating polynomial, increasing the mesh density or a combination of the two, can increase the accuracy of the approximation.

2.3.3 Method of Lines

Numerical solutions to partial differential equations with a time derivative are obtained using the method of lines (MOL) [77]. The idea behind the method of lines is to discretize the spatial derivatives in the PDE using methods such as finite volume or finite element, leaving behind a system of initial value problems (IVPs). The spatial derivatives are now algebraic expressions and solution methods for IVPs can be used to solve the PDE system.

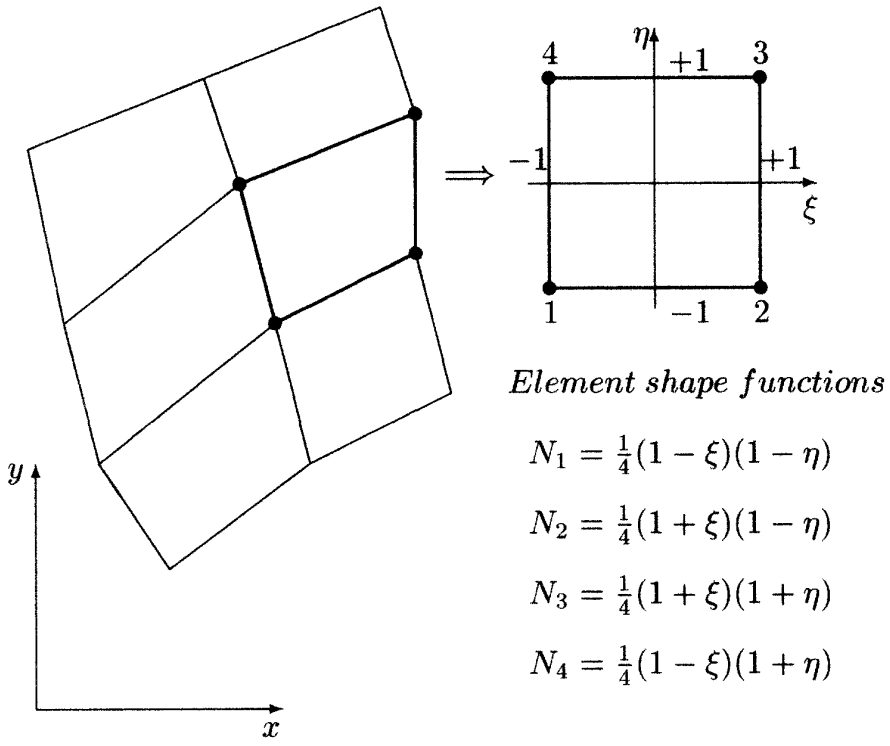


Figure 2.6: Quadrilateral elements and element shape functions [76].

At each point in space, there is a "line" of values, corresponding to the solution in the time domain (Fig. 2.7). The lines combine to form the solution of the PDE over time and space. An example of the method of lines is discussed below.

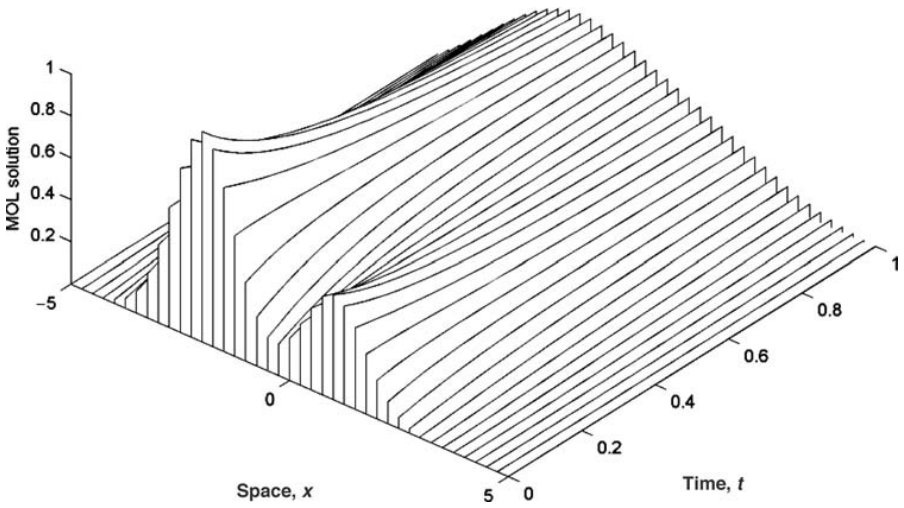


Figure 2.7: Method of lines solution [77].

Using an example of a transient version of the Poisson problem in Eqn. (2.18):

$$\frac{\partial y}{\partial t} = \nabla^2 y + f, \tag{2.21}$$

where the initial condition is $y(x, t = 0) = 0$. The spatial derivative can be discretized in the same manner as that in Section 2.3.2 or using some other spatial discretization method. Let D denote the discretized Laplacian operator, Eqn. (2.21) then becomes:

$$\frac{\mathrm{d}y_i}{\mathrm{d}t} = D(y_i) + f, \tag{2.22}$$

where the time derivative is no longer a partial derivative and y_i is the trial solution at the i-th point in the discretized domain. This IVP can be solved using any of the established

IVP solution methods such as explicit or implicit Euler:

$$\frac{y_i^{n+1} - y_i^n}{\Delta t} = D(y_i^n) + f^n \qquad \text{explicit Euler,} \tag{2.23}$$

$$\frac{y_i^{n+1} - y_i^n}{\Delta t} = D(y_i^{n+1}) + f^{n+1} \qquad \text{implicit Euler.} \tag{2.24}$$

2.3.4 Solution Methods for Single-Phase Incompressible Navier-Stokes Equations

The focus of this work will be on flow systems where the fluid is incompressible. Unlike the case in compressible flows where the pressure can be determined from an equation of state that is a function of density and temperature [78], the density in incompressible flows is constant. The Navier-Stokes equations for single-phase incompressible flow are as follows [79]:

$$\rho \frac{\partial \boldsymbol{v}}{\partial t} + \rho \boldsymbol{\nabla} \cdot (\boldsymbol{v}\boldsymbol{v}) = -\boldsymbol{\nabla} P + \boldsymbol{\nabla} \cdot \boldsymbol{\tau} + \boldsymbol{f}, \tag{2.25}$$

$$\boldsymbol{\nabla} \cdot \boldsymbol{v} = 0. \tag{2.26}$$

This poses a challenge for researchers to develop solution methods that account for the pressure-velocity coupling. An overview of the four families of solution methods for pressure-velocity coupling will be provided in this section.

Artificial Compressibility

The artificial compressibility method was also developed by Chorin [80, 81]. The method is based on the idea that the density in the governing equations is now an artificial density. The pressure is then described through an equation of state that is a function of the artificial density and artificial compressibility [80]:

$$P = \frac{\rho}{\delta}, \tag{2.27}$$

where δ is the artificial compressibility. The artificial compressibility is obtained from the artificial sound speed [80]:

$$c = \frac{1}{\delta^{1/2}}, \tag{2.28}$$

where c is the artificial sound speed. The choice of c is essential as it affects the performance of the numerical method. Since the artificial density is no longer constant, the mass conservation equation becomes [80]:

$$\frac{\partial \rho}{\partial t} + \boldsymbol{\nabla} \cdot \boldsymbol{v} = 0, \tag{2.29}$$

where the time derivative of the artificial density is equal to the mass conservation equation if the flow is incompressible. Substituting in Eqns. (2.27) and (2.28):

$$\frac{1}{c^2}\frac{\partial P}{\partial t} + \boldsymbol{\nabla} \cdot \boldsymbol{v} = 0. \tag{2.30}$$

This method is generally used for computing steady-state solutions since the time evolution of the pressure field is not necessarily accurate but the steady-state value is.

Projection

The projection method was first developed by Chorin [81, 82] for computing transient solutions to the Navier-Stokes equations. The method involves computing a tentative velocity, $\boldsymbol{v}^*$, either implicitly or explicitly from the pressure at the previous time step [76]:

$$\rho\frac{\boldsymbol{v}^* - \boldsymbol{v}^n}{\Delta t} + \rho\boldsymbol{v}^* \cdot \boldsymbol{\nabla}\boldsymbol{v}^* = -\boldsymbol{\nabla}P^n + \boldsymbol{\nabla} \cdot \boldsymbol{\tau}^* + \boldsymbol{f}^*, \qquad \text{implicit,} \tag{2.31}$$

$$\rho\frac{\boldsymbol{v}^* - \boldsymbol{v}^n}{\Delta t} + \rho\boldsymbol{v}^n \cdot \boldsymbol{\nabla}\boldsymbol{v}^n = -\boldsymbol{\nabla}P^n + \boldsymbol{\nabla} \cdot \boldsymbol{\tau}^n + \boldsymbol{f}^n, \qquad \text{explicit.} \tag{2.32}$$

This tentative velocity is not divergence-free and will not satisfy the conservation of mass. The pressure at the next time step is computed from the pressure Poisson equation that

is obtained by taking the divergence of Eqn. (2.34):

$$\nabla^2 P^{n+1} = \rho \frac{\boldsymbol{\nabla} \cdot \boldsymbol{v}^*}{\Delta t}. \tag{2.33}$$

The velocity at the next time step is computed from the following equation [76]:

$$\rho \frac{\boldsymbol{v}^{n+1} - \boldsymbol{v}^*}{\Delta t} + \boldsymbol{\nabla} P^{n+1} = 0. \tag{2.34}$$

The main benefits of this method are that the method is general and does not require arbitrary parameters. The equations do not change when moving from two-dimensions to three-dimensions, which is not the case in some methods.

SIMPLE

The Semi-Implicit Method for Pressure-Linked Equations (SIMPLE) was originally developed for staggered grids by Patankar and Spalding [83]. This method is essentially a predictor-corrector method to obtain the steady-state solution of the Navier-Stokes equations where an initial pressure field, P^*, is provided [83]. The initial pressure field is used to obtain a first approximation of the velocity field, $\boldsymbol{v}^*$, by solving the momentum equation in every dimension. The discretized momentum equation used to solve for $\boldsymbol{v}^*$ will depend on the numerical and discretization methods which are not discussed here. The correct pressure and velocity fields are [78]:

$$P = P^* + P', \tag{2.35a}$$

$$\boldsymbol{v} = \boldsymbol{v}^* + \boldsymbol{v}', \tag{2.35b}$$

where the primes denote correction values. Eqn. (2.35) is substituted into the momentum and mass conservation equations to obtain an expression for the correction values. The contribution of $\boldsymbol{v}'$ from neighboring nodes is omitted from the velocity correction [78]. The pressure update is obtained by substituting Eqn. (2.35) into Eqn. (2.26) which yields the

pressure Poisson equation in terms of P' [83]:

$$P'_i = \sum_{nb} a_{nb} P'_{nb} + R^m_i, \tag{2.36}$$

where nb denotes the neighboring nodes, a is the coefficient and R^m_i is the residual from evaluating Eqn. (2.26). The corrections in Eqn. (2.35) need to be relaxed to improve the convergence of the solution [78]. The algorithm is repeated until convergence is reached. This method is a variation of the projection method previously presented with the pressure correction being computed instead of the pressure at each iteration.

PISO

The PISO (Pressure-Implicit with Splitting of Operators) method was developed for transient simulations by Issa [84]. It is very similar to the SIMPLE method with the main exception aside from it being a transient method being the additional correction step. PISO performs a fixed number of pressure and velocity correction steps within a time step [84]. Similar to the projection method, the incremental value in each prediction-correction step does not satisfy Eqn. (2.26) [84]. The first prediction is obtained in the same manner as in the projection method where $\boldsymbol{\nabla} \cdot \boldsymbol{v}^* \neq 0$. The incremental velocity and pressure values can be generalized as follows [84]:

$$\rho \frac{\boldsymbol{v}^i - \boldsymbol{v}^n}{\Delta t} + \rho \boldsymbol{v}^{i-1} \cdot \boldsymbol{\nabla} \boldsymbol{v}^{i-1} = \boldsymbol{\nabla} \cdot \boldsymbol{\sigma}^{i-1} + \boldsymbol{f}^{n+1}, \tag{2.37}$$

$$\boldsymbol{\nabla}^2 P^{i-1} = \boldsymbol{\nabla} \cdot \left(\boldsymbol{\nabla} \cdot \boldsymbol{\tau}^{i-1} - \rho \boldsymbol{v}^{i-1} \cdot \boldsymbol{\nabla} \boldsymbol{v}^{i-1} \right) + \boldsymbol{\nabla} \cdot \boldsymbol{f}^{n+1} + \frac{\rho \boldsymbol{\nabla} \cdot \boldsymbol{v}^n}{\Delta t}, \tag{2.38}$$

where i refers to the current correction step. Unlike SIMPLE, it does not require relaxation in the pressure correction step [85]. The prediction and correction steps can be repeated as many times as needed to satisfy a convergence criterion. In practice, Issa [84] found that the two correction steps are sufficient.

Chapter 3

Literature Review

In the previous chapter, the theoretical background of the two-fluid model and the numerical method for the solution of flow problems are discussed. In this chapter, pertinent literature on numerical solution to TFM and associated complexities (Section 3.1) and approaches to imposing solid boundaries in multiphase flow (Section 3.2) are reviewed.

3.1 Solution Methods for the Two-Fluid Model

Solution methods for TFM are based on the methods developed for the Navier-Stokes equations (Section 2.3.4). The solution procedure is similar to that of single-phase flow but with the addition of a second momentum equation and a more complex conservation of mass equation. In this section, the different solution methods for TFM reported in the literature are reviewed. This also includes the discussion of the well-posedness of TFM and techniques to ensure phase fraction boundedness in TFM.

Extensions to both (transient) SIMPLE [17, 86] and PISO [87, 88] methods were made to accommodate the governing equations in the two-fluid model. The two-phase extensions of SIMPLE and PISO are commonly found in solvers using the finite volume method [87, 89]. The assumption is made that the two fluids share the same pressure field, thus there is only one pressure unknown. Explicit methods were used for the temporal discretization

of the governing equations. The phase fractions are first computed using values from the previous time step. The pressure and velocities are computed using SIMPLE or PISO with the phase fractions held constant [87]. The phase fractions are later updated with the new pressure and velocities using Eqn. (2.1).

The projection and artificial compressibility methods have also been extended to the two-fluid model and solved using finite element [90, 91]. In the projection method, the incremental velocity is obtained using pressure and phase fractions from the previous time step semi-implicitly, where the relative velocity in the interfacial coupling term is at $n+1$. The pressure is computed using the pressure Poisson equation that is now formulated using the sum of the momentum equations from the phases. The velocity is updated with the new pressure but the phase fractions are still from the previous time step. Finally, the phase fractions are updated using Eqn. (2.1) using the new velocities [90].

The artificial compressibility approach used by Giordano and Magi [91] is also an explicit method. The pressure equation is now formulated using divergence term obtained from when the mass conservation equations are summed together and the phasic density is constant [91]:

$$\sum_q \boldsymbol{\nabla} \cdot (\alpha_q \boldsymbol{v}_q) = 0. \tag{3.1}$$

The governing equations and the pressure equation are solved in a coupled manner to obtain the solution to the equations.

3.1.1 Phase-Fraction Boundedness in the Two-Fluid Model

An important aspect of the two-fluid model is the boundedness of phase fractions. The sum of phase fractions must equal to one at every point in the simulation domain (equality constraint) and the phase fraction of each individual phase needs to be bounded between zero and one (inequality constraint). Mathematically, the two constraints are represented

as follows:

$$\sum_q \alpha_q = 1, \tag{3.2}$$

$$0 \leq \alpha_q \leq 1. \tag{3.3}$$

Equation (3.2) is inherently satisfied in the two-fluid model when solving the conservation of mass equation for only one of the phases, typically the dispersed phase. The phase fraction of the continuous phase is simply $1 - \alpha_d$. Equation (3.3) is not inherently satisfied and additional steps must be taken to ensure that the individual phase fractions are bounded.

The two-fluid model has been implemented in several commercial software packages [89, 92, 93] that are readily available. Other implementations such as `NEPTUNE_CFD` [94, 95] and `OpenFOAM` [87, 96] also exist. A comparative study of some of the TFM implementations is available in the literature [94, 97]. Some of the packages [87, 89, 93, 98] include details on how phase fraction boundedness is enforced.

The different approaches to maintaining phase fraction boundedness available in literature can be broken down into four categories: thresholding [99], flux limiting [100], artificial diffusion [93] and remapping [87, 101]. The use of phase fraction bounding techniques can have an effect on the fidelity of the solution. In this section, existing approaches in maintaining phase fraction boundedness are reviewed.

Depending on the formulation of the momentum equation used, thresholding can be necessary to avoid issues with division by zero. The value of the phase fraction below a threshold is set to be equal to the threshold, typically a small value. One can also choose to threshold the phase fraction outside the momentum equation, this involves zeroing out negative phase fraction values and setting all values above one to be equal to one after the solving Eqn. (2.1). This is the approach taken in Ref. [92]. Thresholding the phase fraction from Eqn. (2.1) will result in a change in the profile and gradient of the phase fraction.

Oliveira and Issa [101] implemented a two-equation method to satisfy Eqns. (3.2) and (3.3) in their in-house code. Equation (2.1) was discretized using an upwind scheme

and solved for both the dispersed and continuous phases. The upwinding ensured that the lower bound in Eqn. (3.3) is satisfied. The resulting phase fractions were then rescaled by a factor of $1/(\alpha_d^* + \alpha_c^*)$, where α_d^* and α_c^* are the values obtained from the conservation of mass equation, satisfying Eqn. (3.2). Since the rescaled individual phase fractions satisfy the lower inequality bound and Eqn. (3.2), the upper inequality bound is also satisfied.

In previous versions of `OpenFOAM`, Weller [87] reformulated the conservation of mass equation such that the phase fraction can be bounded when the conservative form is used. The dispersed phase velocity $\boldsymbol{v}_d$ was decomposed into mean and relative components [87]:

$$\boldsymbol{v}_d = \alpha_c \boldsymbol{v}_c + \alpha_d \boldsymbol{v}_d + \alpha_c \boldsymbol{v}_r. \tag{3.4}$$

Substituting Eqn. (3.4) into Eqn. (2.1) and dividing by ρ_d:

$$\frac{\partial \alpha_d}{\partial t} + \boldsymbol{\nabla} \cdot [\alpha_d(\alpha_c \boldsymbol{v}_c + \alpha_d \boldsymbol{v}_d)] + \boldsymbol{\nabla} \cdot (\boldsymbol{v}_r \alpha_d \alpha_c) = 0. \tag{3.5}$$

The resulting equation is nonlinear and the boundedness of the solution may be compromised when using a higher order spatial discretization scheme [102]. To solve the equation using an iterative linear solver while maintaining the boundedness of the phase fractions, the phase fraction was remapped using a quadratic equation that is a function of both phase fractions [87]:

$$\alpha_d^* = \frac{1}{2}\left[1 - (1 - \alpha_d)^2 + (1 - \alpha_c)^2\right]. \tag{3.6}$$

In more recent versions of `OpenFOAM`, phase fraction boundedness is ensured through the use of a limiter that is based on flux corrected transport called multidimensional universal limiter for explicit solution (MULES) [100]. MULES allows for the possibility of specifying global minimum and maximum values for a given field [98]. The flux that is used to compute the phase fraction in the finite volume method is adjusted by a correction term, λ, and the resulting phase fraction is bounded.

The artificial diffusion approach adds a diffusion term to the conservation of mass equation that will help regularize the solution. In the commercial package `COMSOL` [93], artificial diffusion is used in the conservation of mass equation of the dispersed phase to

minimize the possibility of a negative phase fraction:

$$\frac{\partial \alpha_d}{\partial t} + \boldsymbol{\nabla} \cdot (\alpha_d \boldsymbol{v}_d) = -\boldsymbol{\nabla} \cdot (-\nu_b \boldsymbol{\nabla} \alpha_d), \tag{3.7}$$

where ν_b is the "barrier" viscosity:

$$\nu_b = \frac{\mu_d}{\rho_d}\Big(\exp\Big[\max\Big(-\frac{\alpha_d}{0.0025}, 0\Big)\Big] - 1\Big). \tag{3.8}$$

The barrier viscosity is nonzero when the phase fraction is a negative value. The artificial diffusion term only minimizes the possibility of a negative phase fraction, it does not guarantee that the phase fraction will be positive. Artificial diffusion can alter the governing equation even if the diffusion term is only active when the phase fraction is negative. When the conservation of momentum equation is solved, the phase fractions are thresholded to be between zero and one to regularize the solution [93]. In `COMSOL`, the convective form of the momentum equation is scaled by the phase fraction, which results in $1/\alpha$ terms in the momentum equation, these terms are also thresholded.

3.1.2 Well-Posedness of the Two-Fluid Model

A mathematical model is considered to be well-posed if all of the following are true [64]:

- a solution exists,
- the solution is uniquely determined and
- the solution depend on the initial and boundary conditions continuously.

The well-posedness of the model can affect the stability and accuracy of the solution. An ill-posed two-fluid model can contain unphysical instabilities and excessive numerical diffusion [103]. The TFM CFD practitioners often neglect the ill-posed nature of the equations as many adhere to the criterion that the mesh/grid size must be larger than the bubble size. The effect of this criterion is that the high frequency oscillations that arise from the ill-posedness are coarsened out at the expense of numerical accuracy [104]. This

criterion is based on the idea that since TFM is an averaged model, the length-scale of the computational mesh/grid should be larger than the length-scale of that it is averaging [105]. However, TFM is a continuum model and it should not be affected by the mesh/grid size if well-posed. Thus, it is important to understand the factors that can improve the well-posedness of the two-fluid model.

Drew and Passman [64] studied the well-posedness of the two-fluid model with the simplest form of TFM:

$$\frac{\partial(\alpha_q \rho_q \boldsymbol{v}_q)}{\partial t} + \boldsymbol{\nabla} \cdot (\alpha_q \rho_q \boldsymbol{v}_q \boldsymbol{v}_q) = -\alpha_q \boldsymbol{\nabla} P + \boldsymbol{M}_{q,drag}, \tag{3.9}$$

where the phasic pressures are assumed to be in equilibrium, the viscous stress effects are neglected and the interphase drag is an algebraic expression. This equation is a first-order partial differential equation whose eigenvalues are infinite and complex unless $\boldsymbol{v}_c = \boldsymbol{v}_d$ [64]. The authors also explored the case where the viscous stress of the continuous phase is taken into account. The viscous stress of the continuous phase was treated as a separate variable in the characteristic analysis. From this analysis, the authors concluded that the equations are well-posed for cases where $\boldsymbol{v}_c \neq \boldsymbol{v}_d$ given $\mu_c > 0$, which are most practical cases.

In cases where the viscous stress is either neglected (e.g. inviscid flow) or approximated by an algebraic expression (e.g. algebraic expression for wall stress), the two-fluid model is still ill-posed. Numerous attempts have been made to make the two-fluid model well-posed, including adding the contribution of virtual mass to the momentum exchange [106–108], using interfacial pressure in the governing equations [106, 108–110], adding a momentum flux [111], adding a turbulent dispersion contribution [104] and adding a collision force [112] to the momentum equation. The reported effects of some of the approaches have been contradictory [103] and often dependent on the constants (e.g. virtual mass coefficient, interfacial pressure coefficient, *etc.*) used in the added physics [106, 113, 114].

However, it is important to note that recent work on this topic appears to contradict the conclusion drawn by Drew and Passman [64] that accounting for the viscous stress in the continuous phase is sufficient for the model to become well-posed. Vaidheeswaran et al.

[112] performed a characteristic analysis on the two-fluid model with the viscous stresses of both phases taken into account along with virtual mass, interfacial pressure and collision force. In this case, the viscous stress is considered as the second derivative of the velocity, not a separate variable like in the work by Drew and Passman [64]. The authors found that the two-fluid model is only well-posed up to $\alpha_d \leq 0.26$ when virtual mass and interfacial pressure are considered, which is in agreement with the work carried out by Pauchon and Banerjee [72], and unconditional well-posedness is only achieved when a collision force is added. Additionally, Vaidheeswaran [104] and López de Bertodano et al. [105] stated that the mesh/grid size criterion is unnecessary when the model is well-posed and can result in a loss in accuracy of the solution.

3.2 Embedded Domain Methods for Multiphase Flows

Physical boundaries in multiphase flow systems can be prescribed using an embedded domain method where the information about the physical boundaries is stored independently from the mesh. Embedded domain methods for multiphase flow systems reported in the literature have been largely limited to the immersed boundary method with a few studies conducted using the diffuse-interface method. This section will provide a review of the different studies involving embedded domain methods for multiphase flows.

3.2.1 Immersed Boundary Method

Immersed boundary (IB) methods are methods that treat the solid boundary as being "immersed" inside the simulation grid. They allow for a fixed grid to be utilized for the simulation domain and do not require the mesh to be regularized as the solid boundary moves [50] like in arbitrary Lagrangian-Eulerian methods [48]. The term "immersed boundary" was first coined by Peskin [115] and is often associated with Peskin-type methods where a forcing term is introduced to impose the solid boundary and the location of

the solid boundary is tracked in the Lagrangian frame [50, 51]:

$$\rho\frac{\partial \boldsymbol{v}}{\partial t} + \rho\boldsymbol{v}\cdot\boldsymbol{\nabla}\boldsymbol{v} = -\boldsymbol{\nabla}P + \mu\nabla^2\boldsymbol{v} + \boldsymbol{f} + \boldsymbol{f}_{fluid-solid}. \tag{3.10}$$

However, other methods such as the Navier-Stokes/Brinkman equations [116] are also considered immersed boundary methods. The literature on immersed boundary methods is vast (see Refs. [50, 51] and references therein) and often pertains to single-phase flow systems. In this section, the scope is limited to studies using the immersed boundary method to impose solid boundaries in multiphase fluid flow systems.

Solid Boundary without Flow-Induced Movement

Studies involving immersed boundary methods for multiphase flows in the literature are conducted primarily on multiphase systems with few interfaces where the solid boundary is either stationary or is moving at a known velocity. Systems without flow-induced movement are found in studies of wave propagation [117–123], with applications to ship hydrodynamics and oceanography, injectors [124, 125], porous media [126], hydroplaning [127], *etc.* This type of flow is modeled using interface-capturing methods such as volume-of-fluid (VOF) [34], level-set (LS) [128] and constrained interpolation profile (CIP) [129]. The use of an interface capturing method means that the momentum equation is of the same form as the Navier-Stokes equations but with a source term to account for surface tension. This allows for existing solution techniques to the Navier-Stokes equations to be used to solve the momentum equation.

In the studies reviewed in this section, solid boundary was identified using an indicator function that can either be smooth [117–123, 125, 126, 130–132] or discrete [124, 127, 133]. Son [130] introduced a solid phase fraction, α, that is zero inside the solid, one in the fluid and between zero and one near the fluid-solid interface. The distance away from the fluid-solid interface was used to determine a phase fraction. The transition between fluid to solid was described using a smoothed Heaviside function. The governing equations were

weighted by α [130]:

$$\boldsymbol{\nabla} \cdot (\alpha \boldsymbol{v}) = 0 \qquad \text{for all } \alpha, \tag{3.11a}$$

$$\alpha\rho\frac{\partial \boldsymbol{v}}{\partial t} + \alpha\rho\boldsymbol{v} \cdot \boldsymbol{\nabla}\boldsymbol{v} = -\alpha\boldsymbol{\nabla}P + \boldsymbol{\nabla} \cdot \left(\frac{\mu}{\alpha}\boldsymbol{\nabla}\boldsymbol{v}\right) + \alpha\boldsymbol{f} \qquad \alpha > 0, \tag{3.11b}$$

$$\boldsymbol{v} = 0 \qquad \alpha = 0. \tag{3.11c}$$

Equation (3.11) is valid for the case where the immersed boundary is stationary. This method was validated with simulations of bubbles adhering to a cylindrical solid [130] and the results were found to be comparable to the exact solution for simulations with different contact angles. The immersed boundary treatment was modified to use discontinuous phase fractions and was used to simulate a piezoelectric inkjet process in a subsequent work from Suh and Son [124]. The nozzle geometry was defined using the immersed boundary and the results were found to be in good agreement with other numerical studies in the literature.

Arienti and Sussman [125] studied multiphase diesel injectors with the combined level-set volume-of-fluid/immersed boundary method. The immersed boundary was taken into account in the pressure Poisson equation (see Section 2.3.4) with a condition that the tentative velocity is equal to the solid velocity at the solid faces. This condition was also maintained in the velocity update step. Figure 3.1 shows the pressure distribution from the simulation of a diesel injector. The injector geometry was specified using the immersed boundary method. The velocity profiles at different cross-sections obtained from simulations were compared to experimental data for the same type of diesel injector. Unlike the results from the validation cases where the flow and geometry were much simpler, the results from the diesel injector study only showed some qualitative agreement with the experimental results.

Immersed boundary method was also used in studies involving porous media. Patel et al. [126] combined an existing immersed boundary method with volume-of-fluid to model water flooding in porous media for enhanced oil recovery applications. The method was first validated with simulations of an oil droplet in water on a spherical solid. The droplet shape at different contact angles was used as the basis for comparison between simulation results and the analytical solution. The droplet shapes were in excellent agreement with the

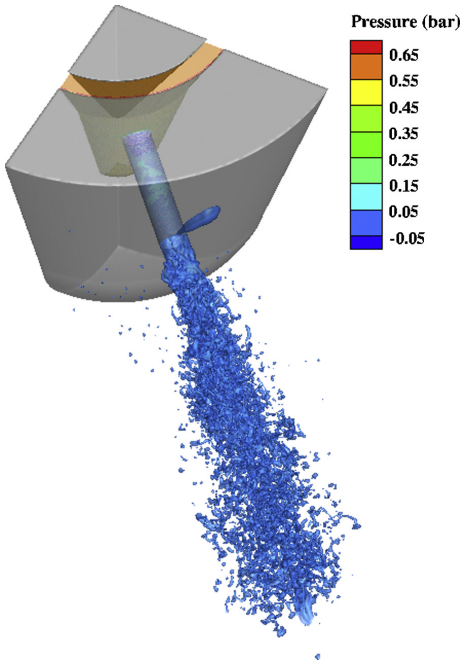


Figure 3.1: Multiphase simulation of a Diesel injector using the immersed boundary method [125].

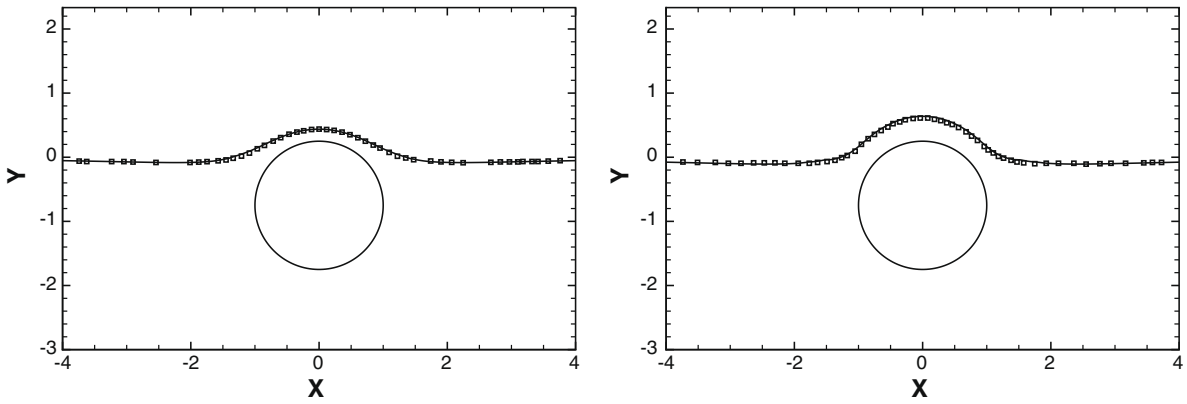
analytical solution. Water flooding in an oil-water system with different porous medium configurations was simulated to determine the amount of residual oil.

Another application of immersed boundary method for multiphase flows is in modeling free surface interactions with solid structures. A VOF/IB method was developed by Shen and Chan [117–119] to study fluid-structure interaction and wave propagation/generation with stationary and moving solid boundaries. A forcing term was added to the momentum equation (Eqn. (3.10)) to impose the solid boundary and the term was treated as diffuse with respect to its influence on the fluid's momentum. The simulations were validated using experimental results and the profiles of the free surface were found to be in agreement. However, the simulated systems in Refs. [117–119] are all for the case where the solid is fully immersed in the liquid and the gas is never in contact with the solid. A similar method was used by Zhang and co-workers [120, 121] to model fluid sloshing in a horizontally-agitated tank [120] and a rotating ellipse in a tank [121]. Simulations were performed to reduce the wave height in the horizontally-agitated tank by studying different baffle configurations.

A variation of the VOF/IB method was used by Gsell et al. [122] to study propagating waves over a complex bottom. The transition from solid to fluid was described using a hyperbolic tangent function. The behavior of a solitary wave as it hits the shore was modeled using the VOF/IB method for different shore configurations. The profile of the free surface near the shore was compared to experimental results and simulation results where the simulation domain is described using a body-fitted mesh. Results experiments and simulations were found to be in good agreement with each other.

Wave-body interactions in ship hydrodynamics were studied by Yang and Stern [123] using a level-set/immersed boundary method with a forcing term. The method was first validated using simulations of water entry and exit. The results were compared to existing numerical results and the results were found to be in good agreement (Fig. 3.2). The model was then used to simulate ship hydrodynamics where a ship is moving at a fixed velocity in water (Fig. 3.3), good agreement was found near the immersed boundary due to higher grid resolution near the boundary.

The immersed boundary method with direct forcing had also been used to study the wetting process of solid particles [131]. The constrained interpolation profile method was



(a) $T = 0.4$ (b) $T = 0.6$

Figure 3.2: Comparison of free surface profiles at different times in Ref. [123]. – results from Ref. [123] □ reference simulation.

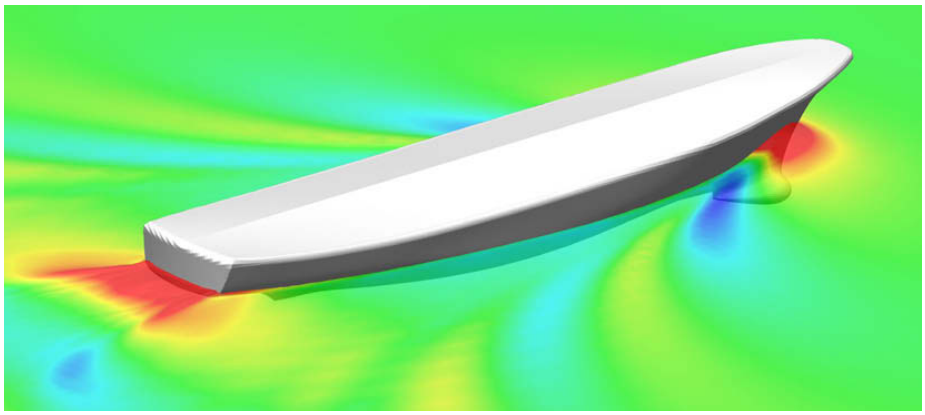


Figure 3.3: Free surface elevation from ship hydrodynamics simulation in Ref. [123]

used to describe the fluid-fluid interaction and a forcing term was added to the momentum equation. The deformation pattern obtained from numerical simulation of droplet impingement on a solid wall was found to be in qualitative agreement with experimental observations.

An alternative to the direct forcing approach for accounting for immersed solid boundaries is to add a penalization term to the momentum equation [50]. The Brinkman equation from porous media was used to penalize the velocity when the solid is present [116]. The solid was treated as a porous medium with a very low permeability. This approach had been used by Horgue et al. [133] and Vincent et al. [127] to study capillary and hydroplaning flows, respectively. In both studies, the solid was identified using a discrete mask function. The solid boundary corresponds to the boundaries of cells that are fully immersed. In order to have an accurate representation of the solid boundary with this approach, a fine mesh grid is required.

Gas-liquid-solid flow inside a rotating drum was modeled using a discrete element method/volume-of-fluid (DEM-VOF) method with an immersed boundary by Sun and Sakai [132]. The discrete element method was used to track the particles inside the physical domain and volume-of-fluid is used to resolve the gas-liquid interface. The solid boundary was imposed using a forcing term. The simulation was validated by comparing the particle bed width and height with experimental results (Fig. 3.4). The particle bed width and height were within 2% relative error of each other. In their later work [134], the volume-of-fluid/immersed boundary method was extended to simulate more complex geometries. The system was a twin-screw kneader with the screw moving with a fixed rotation speed. The simulation results showed qualitative agreement with experimental results.

Solid Boundaries with Flow-Induced Movement

In the event that the solid is moving due forces exerted by the fluid, its motion is determined by solving the Lagrangian form of Newton's second law [50, 51]:

$$\rho_s \frac{\partial^2 \boldsymbol{X}}{\partial t^2} = \boldsymbol{\nabla} \cdot \boldsymbol{\sigma}_s + \boldsymbol{F}_s, \tag{3.12}$$

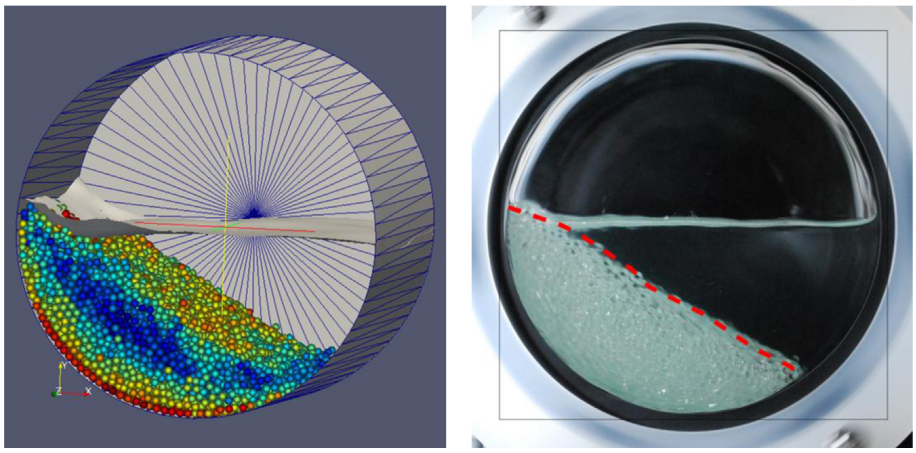


Figure 3.4: Gas-liquid-solid rotating drum. (left) Simulation results (right) experimental results, dashed line indicate computed bed shape from simulations [132].

where ρ_s is the density of the solid, $\boldsymbol{X}$ is the coordinates in the Lagrangian frame, $\boldsymbol{\sigma}_s$ is stress tensor for the solid and $\boldsymbol{F}_s$ is the external force on the solid. From this, the translational and rotational velocities of the solid are computed and the solid velocity at each point is computed from the following expression [135]:

$$\boldsymbol{v}_s = \boldsymbol{v}_{s,t} + \boldsymbol{\omega}_{s,r} \times \boldsymbol{r}, \tag{3.13}$$

where $\boldsymbol{v}_{s,t}$ is the solid translational velocity, $\boldsymbol{\omega}_{s,r}$ is the solid rotational velocity and $\boldsymbol{r}$ is the distance between the solid point and the center of the solid. This solid velocity is used to obtain the forcing term in the immersed boundary method with direct forcing.

Deen et al. [135] used a front-tracking/immersed boundary method to study the interaction between gas bubbles and solid spherical particles. Simulations of a gas bubble rising in a suspension of solid particles were performed to study how the presence of the solid particles affects the terminal rise velocity of the bubble. The surface tension of bubble was high enough that the particles will not penetrate the bubble. It was found that the solid particles significantly decreased the terminal velocity of the bubble. The immersed boundary method was also used to model the behavior of solid objects at the free surface such as a wedge [136, 137] or a barge [137]. These type of studies are particularly useful

in oceanography where free surfaces are prevalent.

3.2.2 Diffuse-Interface Method

The immersed boundary methods discussed in the previous section all treat the solid boundary as a sharp interface. A solid fraction may be used to smear the interface when computing the forcing term, but the interface itself remained sharp. This requires for the velocity to be interpolated to obtain the velocity at a particular node of interest at every time step, which can be costly in large-scale simulations. An alternative to this is to use a diffuse-interface to describe the solid boundary. The diffuse-interface enables the use of the Eulerian frame for both the solid and fluid.

Patel and Natarajan [57] developed a diffuse-interface immersed boundary method for multiphase systems where the fluid was modeled using the volume-of-fluid method. The solid velocity boundary condition was imposed by blending the fluid conservation of momentum equation with the solid velocity. The fluid-solid interface is diffuse over several cells. Validation of the method was performed for different systems including the motion of a rectangular barge, water entry of a circular cylinder and sedimentation of circular particles. The results were in good agreement with other numerical solutions of the same system and exhibited qualitative agreement with experimental results.

A diffuse-domain method was developed by Aland et al. [54] where both the fluid and solid-fluid interfaces were described using diffuse-interfaces. The fluid-fluid interface was resolved using the Cahn-Hilliard equation, which, unlike interface-capturing schemes (volume-of-fluid, level-set, *etc.*), is thermodynamically consistent. Another phase-field was used to describe the solid-fluid interface. The Neumann boundary conditions in the system were imposed using the gradient of the solid-fluid phase-field as the normal vector and Dirichlet boundary conditions were specified using a penalty method. This method was used to simulate various cases of solid interacting with multiphase flow including droplet sliding down a ramp, solid impacting on a liquid-liquid interface (Fig. 3.5), droplet moving in a serpentine channel and flow field in thin film growth by electrodeposition.

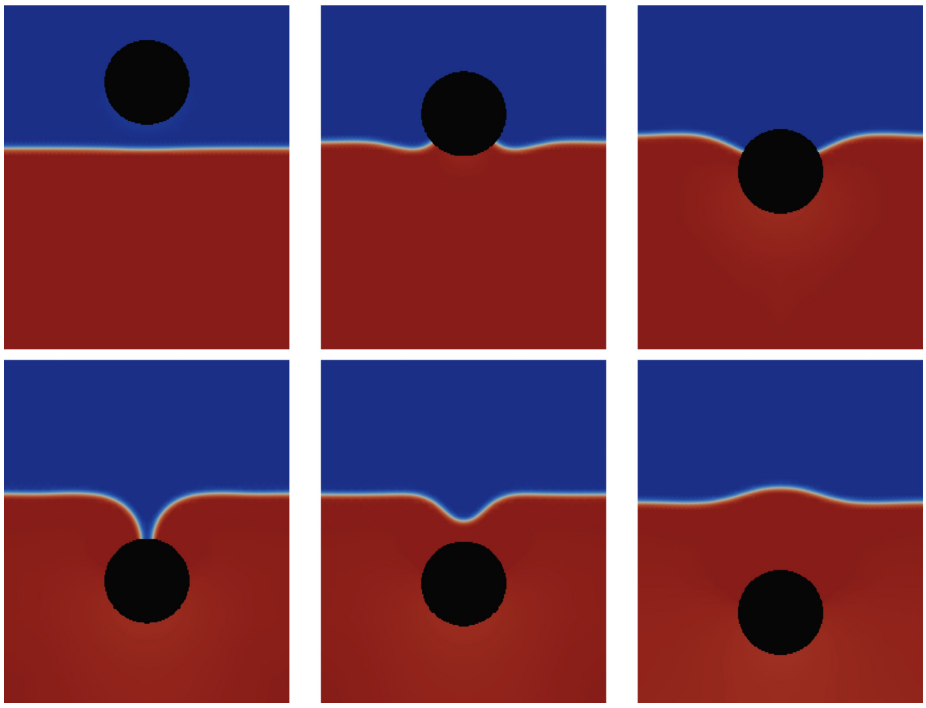


Figure 3.5: Solid ball impacting into a liquid-liquid interface [54].

Chapter 4

Finite Element Formulation for the Solution of the Two-Fluid Model

Prior to developing a diffuse-interface method for imposing solid boundaries in multiphase flow, a solver for the two-fluid model equations is developed. In this work, the governing equations for the two-fluid model that were presented in Section 2.2 are solved using the finite element method. The finite element method requires the weak formulation of the governing equations, which are presented in this chapter. The time discretization and adaptive time-stepping scheme used in this work are also introduced in this chapter.

4.1 Modified Incremental Pressure Correction Scheme

From Section 2.2, the governing equations for the two-fluid model are as follows:

$$\frac{\partial(\alpha_q\overline{\overline{\rho_q}})}{\partial t} + \boldsymbol{\nabla}\cdot\left(\alpha_q\overline{\overline{\rho_q}}\widehat{\boldsymbol{v}_q}\right) = 0, \tag{2.1}$$

$$\frac{\partial\left(\alpha_c\overline{\overline{\rho_c}}\widehat{\boldsymbol{v}}_c\right)}{\partial t}+\boldsymbol{\nabla}\cdot\left(\alpha_c\overline{\overline{\rho_c}}\widehat{\boldsymbol{v}}_c\widehat{\boldsymbol{v}}_c\right)=-\alpha_c\boldsymbol{\nabla}\overline{\overline{P_c}}+\boldsymbol{\nabla}\cdot\left(\alpha_c\overline{\overline{\boldsymbol{\tau}_c}}\right)+\alpha_c\overline{\overline{\rho_c}}\widehat{\boldsymbol{g}}+\boldsymbol{M}_c+\left(\overline{\overline{P_{int}}}-\overline{\overline{P_c}}\right)\boldsymbol{\nabla}\alpha_c, \tag{2.4a}$$

$$\frac{\partial\left(\alpha_d\overline{\overline{\rho_d}}\widehat{\boldsymbol{v}}_d\right)}{\partial t}+\boldsymbol{\nabla}\cdot\left(\alpha_d\overline{\overline{\rho_d}}\widehat{\boldsymbol{v}}_d\widehat{\boldsymbol{v}}_d\right)=-\alpha_d\boldsymbol{\nabla}\overline{\overline{P_{int}}}+\boldsymbol{\nabla}\cdot\left(\alpha_d\overline{\overline{\boldsymbol{\tau}_d}}\right)+\alpha_d\overline{\overline{\rho_d}}\widehat{\boldsymbol{g}}+\boldsymbol{M}_d. \tag{2.4b}$$

In this work, only the interphase momentum transfer due to drag is considered. The fluid-fluid system of interest is a gas-liquid system where liquid, l, is the continuous phase and gas, g, is the dispersed phase, the governing equations will reflect this from hereon in. Substituting Eqns. (2.8) and (2.12) into (2.4) and dropping the averaging notation:

$$\frac{\partial(\alpha_l\rho_l\boldsymbol{v}_l)}{\partial t}+\boldsymbol{\nabla}\cdot(\alpha_l\rho_l\boldsymbol{v}_l\boldsymbol{v}_l)=-\alpha_l\boldsymbol{\nabla}P_l+\boldsymbol{\nabla}\cdot(\alpha_l\boldsymbol{\tau}_l)+\alpha_l\rho_l\boldsymbol{g}+\frac{3}{4}\alpha_g\rho_l\frac{C_D}{d_b}\|\boldsymbol{v}_r\|\boldsymbol{v}_r-C_p\rho_l\boldsymbol{v}_r\cdot\boldsymbol{v}_r\boldsymbol{\nabla}\alpha_l, \tag{4.1a}$$

$$\frac{\partial(\alpha_g\rho_g\boldsymbol{v}_g)}{\partial t}+\boldsymbol{\nabla}\cdot(\alpha_g\rho_g\boldsymbol{v}_g\boldsymbol{v}_g)=-\alpha_g\boldsymbol{\nabla}(P_l-C_p\rho_l\boldsymbol{v}_r\cdot\boldsymbol{v}_r)+\boldsymbol{\nabla}\cdot(\alpha_g\boldsymbol{\tau}_g)+\alpha_g\rho_g\boldsymbol{g}-\frac{3}{4}\alpha_g\rho_l\frac{C_D}{d_b}\|\boldsymbol{v}_r\|\boldsymbol{v}_r. \tag{4.1b}$$

The both the gas and liquid phases are assumed to be Newtonian fluids, thus the viscous stress tensor is $\boldsymbol{\tau}_q=\mu_q\left(\boldsymbol{\nabla}\boldsymbol{v}_q+\boldsymbol{\nabla}\boldsymbol{v}_q^T\right)$. For compactness, $\boldsymbol{\tau}_q$ is retained in the subsequent equations.

The conservation of momentum equations defined in Eqn. (4.1) are the conservative form, but their solution becomes degenerate as the phase fraction approaches zero. To avoid this issue, the dimensionless convective form of the momentum equations, scaled by $\alpha_q\rho_q$, is used instead. The dimensionless quantities are defined as follows: $\tilde{\boldsymbol{v}}=\boldsymbol{v}/v_s$, $\tilde{t}=t/t_s$, $\tilde{\boldsymbol{x}}=\boldsymbol{x}/x_s$, $\tilde{P}=(P-P_0)/P_s$, $\tilde{\boldsymbol{g}}=\boldsymbol{g}/g_s$, $\tilde{\boldsymbol{\nabla}}=x_s\boldsymbol{\nabla}$ and $\tilde{d}_b=d_b/x_s$. The parameters v_s, t_s, x_s, P_s and g_s are scaling parameters that represent the characteristic velocity, time, length, pressure and gravitational force scales of the system, respectively. This results in

Table 4.1: Dimensionless groups

Parameter	Expression
Time	$t_s = v_s/x_s$
Pressure	$P_s = \rho_l g_s h$ $P_0 = 0$
Euler number	$Eu_q = P_s/\rho_q v_s^2$
Reynolds number	$Re_q = \rho_q v_s x_s/\mu_q$
Froude number	$Fr = v_s/\sqrt{g_s x_s}$

the following scaled equations:

$$\begin{aligned}\frac{\partial \tilde{\boldsymbol{v}}_l}{\partial \tilde{t}} + \tilde{\boldsymbol{v}}_l \cdot \tilde{\boldsymbol{\nabla}} \tilde{\boldsymbol{v}}_l = &-Eu_l \tilde{\boldsymbol{\nabla}} \tilde{P}_l + \frac{1}{Re_l} \frac{\tilde{\boldsymbol{\nabla}} \alpha_l \cdot \tilde{\boldsymbol{\tau}}_l}{\alpha_l} + \frac{1}{Re_l} \tilde{\boldsymbol{\nabla}} \cdot \tilde{\boldsymbol{\tau}}_l + \frac{1}{Fr^2} \tilde{\boldsymbol{g}} \\ &+ \frac{3}{4} \frac{\alpha_g}{\alpha_l} \frac{C_D}{\tilde{d}_b} \|\tilde{\boldsymbol{v}}_r\| \tilde{\boldsymbol{v}}_r - C_P \tilde{\boldsymbol{v}}_r \cdot \tilde{\boldsymbol{v}}_r \frac{\tilde{\boldsymbol{\nabla}} \alpha_l}{\alpha_l},\end{aligned} \tag{4.2a}$$

$$\begin{aligned}\frac{\partial \tilde{\boldsymbol{v}}_g}{\partial \tilde{t}} + \tilde{\boldsymbol{v}}_g \cdot \tilde{\boldsymbol{\nabla}} \tilde{\boldsymbol{v}}_g = &-Eu_g \tilde{\boldsymbol{\nabla}} \tilde{P}_l + C_P \tilde{\boldsymbol{\nabla}} (\tilde{\boldsymbol{v}}_r \cdot \tilde{\boldsymbol{v}}_r) \frac{\rho_l}{\rho_g} + \frac{1}{Re_g} \frac{\tilde{\boldsymbol{\nabla}} \alpha_g \cdot \tilde{\boldsymbol{\tau}}_g}{\alpha_g} \\ &+ \frac{1}{Re_g} \tilde{\boldsymbol{\nabla}} \cdot \tilde{\boldsymbol{\tau}}_g + \frac{1}{Fr^2} \tilde{\boldsymbol{g}} - \frac{3}{4} \frac{\rho_l}{\rho_g} \frac{C_D}{\tilde{d}_b} \|\tilde{\boldsymbol{v}}_r\| \tilde{\boldsymbol{v}}_r,\end{aligned} \tag{4.2b}$$

$$\frac{\partial \alpha_g}{\partial \tilde{t}} + \tilde{\boldsymbol{\nabla}} \cdot (\alpha_g \tilde{\boldsymbol{v}}_g) = 0, \tag{4.2c}$$

$$\alpha_l = 1 - \alpha_g, \tag{4.2d}$$

where the dimensionless groups are given in Table 4.1 and Eqn. (4.2d) is a result of the equality constraint (Eqn. (3.2)).

The governing equations are solved using a modification of the incremental pressure correction scheme (IPCS) [138]. IPCS was originally developed for single-phase flow and has been shown to be efficient and accurate [139]. In this work, a modification is made to accommodate for the addition of the conservation of mass and second conservation of momentum equation in the two-fluid model. Explicit Euler time discretization is used to demonstrate the form of the discretized equations. There are four steps in the scheme, the first step is to compute a tentative velocity, $\tilde{\boldsymbol{v}}_q^*$, using pressure and velocities from the

previous time step:

$$\frac{\tilde{\boldsymbol{v}}_l^* - \tilde{\boldsymbol{v}}_l^n}{\Delta t} + \tilde{\boldsymbol{v}}_l^n \cdot \tilde{\boldsymbol{\nabla}} \tilde{\boldsymbol{v}}_l^n = -Eu_l \tilde{\boldsymbol{\nabla}} \tilde{P}_l^n + \frac{1}{Re_l} \frac{\tilde{\boldsymbol{\nabla}} \alpha_l^n \cdot \tilde{\boldsymbol{\tau}}_l^{n+\frac{1}{2}}}{\alpha_l^n} + \frac{1}{Re_l} \tilde{\boldsymbol{\nabla}} \cdot \tilde{\boldsymbol{\tau}}_l^{n+\frac{1}{2}} + \frac{1}{Fr^2} \tilde{\boldsymbol{g}} + \frac{3}{4} \frac{\alpha_g^n}{\alpha_l^n} \frac{C_D}{\tilde{d}_b} \|\tilde{\boldsymbol{v}}_r^n\| \tilde{\boldsymbol{v}}_r^n - C_P \tilde{\boldsymbol{v}}_r^n \cdot \tilde{\boldsymbol{v}}_r^n \frac{\tilde{\boldsymbol{\nabla}} \alpha_l^n}{\alpha_l^n} \quad \text{in} \quad \Omega, \tag{4.3a}$$

$$\frac{\tilde{\boldsymbol{v}}_g^* - \tilde{\boldsymbol{v}}_g^n}{\Delta t} + \tilde{\boldsymbol{v}}_g^n \cdot \tilde{\boldsymbol{\nabla}} \tilde{\boldsymbol{v}}_g^n = -Eu_g \tilde{\boldsymbol{\nabla}} \tilde{P}_l^n + C_P \tilde{\boldsymbol{\nabla}} (\tilde{\boldsymbol{v}}_r^n \cdot \tilde{\boldsymbol{v}}_r^n) \frac{\rho_l}{\rho_g} + \frac{1}{Re_g} \frac{\tilde{\boldsymbol{\nabla}} \alpha_g^n \cdot \tilde{\boldsymbol{\tau}}_g^{n+\frac{1}{2}}}{\alpha_g^n} + \frac{1}{Re_g} \tilde{\boldsymbol{\nabla}} \cdot \tilde{\boldsymbol{\tau}}_g^{n+\frac{1}{2}} + \frac{1}{Fr^2} \tilde{\boldsymbol{g}} - \frac{3}{4} \frac{\rho_l}{\rho_g} \frac{C_D}{\tilde{d}_b} \|\tilde{\boldsymbol{v}}_r^n\| \tilde{\boldsymbol{v}}_r^n \quad \text{in} \quad \Omega, \tag{4.3b}$$

$$\tilde{\boldsymbol{v}}_q^* = \tilde{\boldsymbol{v}}_{q,BC}^{n+1} \quad \text{on} \quad \Gamma_D, \tag{4.3c}$$

$$\boldsymbol{n} \cdot \tilde{\boldsymbol{\tau}}_q^{n+\frac{1}{2}} = \boldsymbol{0} \quad \text{on} \quad \Gamma_N, \tag{4.3d}$$

where $\tilde{\boldsymbol{\tau}}_q^{n+\frac{1}{2}} = \left(\tilde{\boldsymbol{\tau}}_q^* + \tilde{\boldsymbol{\tau}}_q^n\right)/2$. The $\left(\tilde{\boldsymbol{\nabla}} \alpha_q^n\right)/\alpha_q^n$ term is approximated by $\tilde{\boldsymbol{\nabla}} \left(\ln \alpha_q'\right)$ where α_q' is α_q^n thresholded to be above a minimum value, 10^{-5} in this work. Then, the tentative velocity is used to compute an update to the pressure field. Using the conservative form of the momentum equations and the incompressibility criterion of the two-fluid model, $\boldsymbol{\nabla} \cdot \sum_q \alpha_q \boldsymbol{v}_q = 0$, the following pressure Poisson equation is obtained:

$$\tilde{\boldsymbol{\nabla}} \cdot \left[\sum_q Eu_q \alpha_q^n \tilde{\boldsymbol{\nabla}} (\tilde{P}_l^{n+1} - \tilde{P}_l^n) \right] = \tilde{\boldsymbol{\nabla}} \cdot \sum_q \left(\frac{\alpha_q^n \tilde{\boldsymbol{v}}_q^*}{\Delta t} \right) \quad \text{in} \quad \Omega, \tag{4.4a}$$

$$\boldsymbol{n} \cdot \tilde{\boldsymbol{\nabla}} \left(\tilde{P}_l^{n+1} - \tilde{P}_l^n \right) = 0 \quad \text{on} \quad \Gamma_D, \tag{4.4b}$$

$$\tilde{P}_l^{n+1} = \tilde{P}_{l,BC}^{n+1} \quad \text{on} \quad \Gamma_N. \tag{4.4c}$$

This pressure is used to update the velocity fields:

$$\frac{\tilde{\boldsymbol{v}}_l^{n+1} - \tilde{\boldsymbol{v}}_l^*}{\Delta t} = -Eu_l \tilde{\boldsymbol{\nabla}} \left(\tilde{P}_l^{n+1} - \tilde{P}_l^n \right) \qquad \text{in} \quad \Omega, \tag{4.5a}$$

$$\frac{\tilde{\boldsymbol{v}}_g^{n+1} - \tilde{\boldsymbol{v}}_g^*}{\Delta t} = -Eu_g \tilde{\boldsymbol{\nabla}} \left(\tilde{P}_l^{n+1} - \tilde{P}_l^n \right) \qquad \text{in} \quad \Omega, \tag{4.5b}$$

$$\tilde{\boldsymbol{v}}_q^{n+1} = \tilde{\boldsymbol{v}}_{q,BC}^{n+1} \qquad \text{on} \quad \Gamma_D. \tag{4.5c}$$

Finally, the phase fractions are computed using the updated velocity field:

$$\frac{\alpha_g^{n+1} - \alpha_g^n}{\Delta t} + \tilde{\boldsymbol{\nabla}} \cdot \left(\alpha_g^{n+1} \tilde{\boldsymbol{v}}_g^{n+1} \right) = 0 \qquad \text{in} \quad \Omega, \tag{4.6a}$$

$$\alpha_g^{n+1} = \alpha_{g,BC}^{n+1} \qquad \text{on} \quad \Gamma_D, \tag{4.6b}$$

$$\alpha_l^{n+1} = 1 - \alpha_g^{n+1} \qquad \text{in} \quad \Omega. \tag{4.6c}$$

4.2 Weak Form of Governing Equations

The time-discretized equations presented in the previous section are solved using the method of lines. The finite element method is used to approximate the spatial derivatives in the differential equations. FEM requires the differential equations to be formulated into their weak formulations. In this section, weak formulations of Eqns. (4.3a), (4.3b), (4.4a), (4.5a), (4.5b) and (4.6a) with the appropriate boundary conditions are described. The time discretization scheme is still the explicit Euler method but extension to other time discretization schemes is possible.

Tentative Velocity Taking the inner product of Eqns. (4.3a) and (4.3b) with the test function for each phase, $\boldsymbol{\varphi}_q$, and integrating over the domain yields the following:

$$\begin{aligned}\left\langle \frac{\tilde{\boldsymbol{v}}_l^* - \tilde{\boldsymbol{v}}_l^n}{\Delta t}, \boldsymbol{\varphi}_l \right\rangle_\Omega + \left\langle \tilde{\boldsymbol{v}}_l^n \cdot \tilde{\boldsymbol{\nabla}} \tilde{\boldsymbol{v}}_l^n, \boldsymbol{\varphi}_l \right\rangle_\Omega = &-\left\langle Eu_l \tilde{\boldsymbol{\nabla}} \tilde{P}_l^n, \boldsymbol{\varphi}_l \right\rangle_\Omega + \left\langle \frac{1}{Re_l} \frac{\tilde{\boldsymbol{\nabla}} \alpha_l^n \cdot \tilde{\boldsymbol{\tau}}_l^{n+\frac{1}{2}}}{\alpha_l^n}, \boldsymbol{\varphi}_l \right\rangle_\Omega \\ &- \left\langle \frac{1}{Re_l} \tilde{\boldsymbol{\tau}}_l^{n+\frac{1}{2}}, \tilde{\boldsymbol{\nabla}} \boldsymbol{\varphi}_l \right\rangle_\Omega + \left\langle \frac{1}{Re_l} \boldsymbol{n} \cdot \tilde{\boldsymbol{\tau}}_l^{n+\frac{1}{2}}, \boldsymbol{\varphi}_l \right\rangle_{\Gamma_N} \\ &+ \left\langle \frac{1}{Fr^2} \tilde{\boldsymbol{g}}, \boldsymbol{\varphi}_l \right\rangle_\Omega + \left\langle \frac{3}{4} \frac{\alpha_g^n}{\alpha_l^n} \frac{C_D}{\tilde{d}_b} \|\tilde{\boldsymbol{v}}_r^n\| \tilde{\boldsymbol{v}}_r^n, \boldsymbol{\varphi}_l \right\rangle_\Omega \\ &- \left\langle C_P \tilde{\boldsymbol{v}}_r^n \cdot \tilde{\boldsymbol{v}}_r^n \frac{\tilde{\boldsymbol{\nabla}} \alpha_l^n}{\alpha_l^n}, \boldsymbol{\varphi}_l \right\rangle_\Omega,\end{aligned} \tag{4.7a}$$

$$\begin{aligned}\left\langle \frac{\tilde{\boldsymbol{v}}_g^* - \tilde{\boldsymbol{v}}_g^n}{\Delta t}, \boldsymbol{\varphi}_g \right\rangle_\Omega + \left\langle \tilde{\boldsymbol{v}}_g^n \cdot \tilde{\boldsymbol{\nabla}} \tilde{\boldsymbol{v}}_g^n, \boldsymbol{\varphi}_g \right\rangle_\Omega = &-\left\langle Eu_g \tilde{\boldsymbol{\nabla}} \tilde{P}_l^n - C_P \tilde{\boldsymbol{\nabla}} (\tilde{\boldsymbol{v}}_r^n \cdot \tilde{\boldsymbol{v}}_r^n) \frac{\rho_l}{\rho_g}, \boldsymbol{\varphi}_g \right\rangle_\Omega \\ &+ \left\langle \frac{1}{Re_g} \frac{\tilde{\boldsymbol{\nabla}} \alpha_g^n \cdot \tilde{\boldsymbol{\tau}}_g^{n+\frac{1}{2}}}{\alpha_g^n}, \boldsymbol{\varphi}_g \right\rangle_\Omega \\ &- \left\langle \frac{1}{Re_g} \tilde{\boldsymbol{\tau}}_g^{n+\frac{1}{2}}, \tilde{\boldsymbol{\nabla}} \boldsymbol{\varphi}_g \right\rangle_\Omega + \left\langle \frac{1}{Re_g} \boldsymbol{n} \cdot \tilde{\boldsymbol{\tau}}_g^{n+\frac{1}{2}}, \boldsymbol{\varphi}_g \right\rangle_{\Gamma_N} \\ &+ \left\langle \frac{1}{Fr^2} \tilde{\boldsymbol{g}}, \boldsymbol{\varphi}_g \right\rangle_\Omega - \left\langle \frac{3}{4} \frac{\rho_l}{\rho_g} \frac{C_D}{\tilde{d}_b} \|\tilde{\boldsymbol{v}}_r^n\| \tilde{\boldsymbol{v}}_r^n, \boldsymbol{\varphi}_g \right\rangle_\Omega,\end{aligned} \tag{4.7b}$$

where the subscript Ω denotes integral over the entire domain and Γ_N denotes integral over boundaries where the Neumann boundary condition applies. From Eqn. (4.3d), the normal component of the viscous stress tensor is equal to zero, substituting the relationship into

the weak formulation:

$$\begin{aligned}\left\langle \frac{\tilde{\boldsymbol{v}}_l^* - \tilde{\boldsymbol{v}}_l^n}{\Delta t}, \boldsymbol{\varphi}_l \right\rangle_\Omega + \left\langle \tilde{\boldsymbol{v}}_l^n \cdot \tilde{\boldsymbol{\nabla}} \tilde{\boldsymbol{v}}_l^n, \boldsymbol{\varphi}_l \right\rangle_\Omega &= -\left\langle Eu_l \tilde{\boldsymbol{\nabla}} \tilde{P}_l^n, \varphi \right\rangle_\Omega + \left\langle \frac{1}{Re_l} \frac{\tilde{\boldsymbol{\nabla}} \alpha_l^n \cdot \tilde{\boldsymbol{\tau}}_l^{n+\frac{1}{2}}}{\alpha_l^n}, \boldsymbol{\varphi}_l \right\rangle_\Omega \\ &\quad - \left\langle \frac{1}{Re_l} \tilde{\boldsymbol{\tau}}_l^{n+\frac{1}{2}}, \tilde{\boldsymbol{\nabla}} \boldsymbol{\varphi}_l \right\rangle_\Omega + \left\langle \frac{1}{Fr^2} \tilde{\boldsymbol{g}}, \boldsymbol{\varphi}_l \right\rangle_\Omega \\ &\quad + \left\langle \frac{3}{4} \frac{\alpha_g^n}{\alpha_l^n} \frac{C_D}{\tilde{d}_b} \|\tilde{\boldsymbol{v}}_r^n\| \tilde{\boldsymbol{v}}_r^n, \boldsymbol{\varphi}_l \right\rangle_\Omega \\ &\quad - \left\langle C_P \tilde{\boldsymbol{v}}_r^n \cdot \tilde{\boldsymbol{v}}_r^n \frac{\tilde{\boldsymbol{\nabla}} \alpha_l^n}{\alpha_l^n}, \boldsymbol{\varphi}_l \right\rangle_\Omega, \end{aligned} \tag{4.8a}$$

$$\tilde{\boldsymbol{v}}_l^*|_{\Gamma_D} = \tilde{\boldsymbol{v}}_{l,BC}^{n+1}, \tag{4.8b}$$

$$\begin{aligned}\left\langle \frac{\tilde{\boldsymbol{v}}_g^* - \tilde{\boldsymbol{v}}_g^n}{\Delta t}, \boldsymbol{\varphi}_g \right\rangle_\Omega + \left\langle \tilde{\boldsymbol{v}}_g^n \cdot \tilde{\boldsymbol{\nabla}} \tilde{\boldsymbol{v}}_g^n, \boldsymbol{\varphi}_g \right\rangle_\Omega &= -\left\langle Eu_g \tilde{\boldsymbol{\nabla}} \tilde{P}_l^n - C_P \tilde{\boldsymbol{\nabla}} (\tilde{\boldsymbol{v}}_r^n \cdot \tilde{\boldsymbol{v}}_r^n) \frac{\rho_l}{\rho_g}, \boldsymbol{\varphi}_g \right\rangle_\Omega \\ &\quad + \left\langle \frac{1}{Re_g} \frac{\tilde{\boldsymbol{\nabla}} \alpha_g^n \cdot \tilde{\boldsymbol{\tau}}_g^{n+\frac{1}{2}}}{\alpha_g^n}, \boldsymbol{\varphi}_g \right\rangle_\Omega \\ &\quad - \left\langle \frac{1}{Re_g} \tilde{\boldsymbol{\tau}}_g^{n+\frac{1}{2}}, \tilde{\boldsymbol{\nabla}} \boldsymbol{\varphi}_g \right\rangle_\Omega + \left\langle \frac{1}{Fr^2} \tilde{\boldsymbol{g}}, \boldsymbol{\varphi}_g \right\rangle_\Omega \\ &\quad - \left\langle \frac{3}{4} \frac{\rho_l}{\rho_g} \frac{C_D}{\tilde{d}_b} \|\tilde{\boldsymbol{v}}_r^n\| \tilde{\boldsymbol{v}}_r^n, \boldsymbol{\varphi}_g \right\rangle_\Omega, \end{aligned} \tag{4.8c}$$

$$\tilde{\boldsymbol{v}}_g^*|_{\Gamma_D} = \tilde{\boldsymbol{v}}_{g,BC}^{n+1}, \tag{4.8d}$$

where Γ_D denotes the boundaries with Dirichlet boundary conditions.

Pressure The weak formulation of the pressure Poisson equation (Eqn. (4.4a)) with the test function, φ_p is:

$$-\left\langle \sum_q Eu_q \alpha_q^n \tilde{\boldsymbol{\nabla}}(\tilde{P}_l^{n+1} - \tilde{P}_l^n), \tilde{\boldsymbol{\nabla}}\varphi_p \right\rangle_\Omega + \left\langle \sum_q Eu_q \alpha_q^n \boldsymbol{n} \cdot \tilde{\boldsymbol{\nabla}}(\tilde{P}_l^{n+1} - \tilde{P}_l^n), \varphi_p \right\rangle_{\Gamma_D} = \left\langle \tilde{\boldsymbol{\nabla}} \cdot \sum_q \left(\frac{\alpha_q^n \tilde{\boldsymbol{v}}_q^*}{\Delta t} \right), \varphi_p \right\rangle_\Omega . \tag{4.9}$$

Given Eqn. (4.4b), the equation can be rewritten as:

$$-\left\langle \sum_q Eu_q \alpha_q^n \tilde{\boldsymbol{\nabla}}(\tilde{P}_l^{n+1} - \tilde{P}_l^n), \tilde{\boldsymbol{\nabla}}\varphi_p \right\rangle_\Omega = \left\langle \tilde{\boldsymbol{\nabla}} \cdot \sum_q \left(\frac{\alpha_q^n \tilde{\boldsymbol{v}}_q^*}{\Delta t} \right), \varphi_p \right\rangle_\Omega , \tag{4.10a}$$

$$\tilde{P}_l^{n+1}\Big|_{\Gamma_N} = \tilde{P}_{l,BC}^{n+1}. \tag{4.10b}$$

Velocity Update The inner products of Eqns. (4.5a) and (4.5b) with $\boldsymbol{\varphi}_q$ integrated over Ω are:

$$\left\langle \frac{\tilde{\boldsymbol{v}}_l^{n+1} - \tilde{\boldsymbol{v}}_l^*}{\Delta t}, \boldsymbol{\varphi}_l \right\rangle_\Omega = -\left\langle Eu_l \tilde{\boldsymbol{\nabla}}(\tilde{P}_l^{n+1} - \tilde{P}_l^n), \boldsymbol{\varphi}_l \right\rangle_\Omega, \tag{4.11a}$$

$$\tilde{\boldsymbol{v}}_l^{n+1}\Big|_{\Gamma_D} = \tilde{\boldsymbol{v}}_{l,BC}^{n+1}, \tag{4.11b}$$

$$\left\langle \frac{\tilde{\boldsymbol{v}}_g^{n+1} - \tilde{\boldsymbol{v}}_g^*}{\Delta t}, \boldsymbol{\varphi}_g \right\rangle_\Omega = -\left\langle Eu_g \tilde{\boldsymbol{\nabla}}(\tilde{P}_l^{n+1} - \tilde{P}_l^n), \boldsymbol{\varphi}_g \right\rangle_\Omega, \tag{4.11c}$$

$$\tilde{\boldsymbol{v}}_g^{n+1}\Big|_{\Gamma_D} = \tilde{\boldsymbol{v}}_{g,BC}^{n+1}. \tag{4.11d}$$

Phase Fraction Update Lastly, the same procedure is repeated where the inner product of Eqn. (4.6a) with the test function φ_α is integrated over the simulation domain to give

the following weak formulation:

$$\left\langle \frac{\alpha_g^{n+1} - \alpha_g^n}{\Delta t}, \varphi_\alpha \right\rangle_\Omega + \left\langle \tilde{\boldsymbol{\nabla}} \cdot \left(\alpha_g^{n+1} \tilde{\boldsymbol{v}}_g^{n+1}\right), \varphi_\alpha \right\rangle_\Omega = 0, \tag{4.12a}$$

$$\alpha_g^{n+1}|_{\Gamma_D} = \alpha_{g,BC}^{n+1}. \tag{4.12b}$$

4.3 Adaptive Time-Stepping

The choice of time step is crucial to the numerical stability of the solution method and the computational time required to complete a simulation. Adaptive time-stepping is used in this work to select a time step that constrains the local error to a user-specified value while maintaining numerical stability. The local error, l, of the solution obtained from a p-th order method is proportional to the step size, $l \propto (\Delta t)^{p+1}$. The local error at time t^{n+1} is given as:

$$l^{n+1} = \hat{y}^{n+1} - y^{n+1}, \tag{4.13}$$

where $\hat{y}^{n+1}$ is the solution obtained from a higher order method and y^{n+1} is the solution from the p-th order method.

The goal of adaptive time-stepping is to select a step size such that $\|l^{n+1}\| < \epsilon_l$, where ϵ_l is the user-specified tolerance. This is achieved using the following expression [140]:

$$(\Delta t)' = \Delta t \left(\frac{0.9\epsilon_l}{\|l^{n+1}\|} \right)^{\frac{1}{p+1}}, \tag{4.14}$$

where $(\Delta t)'$ is the new step size and 0.9 is a safety factor to increase the likelihood that the new step size will result in a solution that conforms to the local error constraint. When $\|l^{n+1}\| > \epsilon_l$, Eqn. (4.14) will result in a new step size that is smaller than the original step size. If that is the case, the computation is repeated with $(\Delta t)'$ until the local error constraint is satisfied.

In fluid flow problems, an additional constraint in the form of the Courant number is

also required [85]. The Courant number is the ratio of the step size and the characteristic time associated with convection:

$$Co = \frac{v\Delta t}{h}, \tag{4.15}$$

where v is the velocity and h is the mesh element size. In multidimensional systems, this is given as:

$$Co = \sum_i \frac{|v_i|\Delta t}{h}, \tag{4.16}$$

where v_i is the velocity component in the i direction. In multiphase flow, the Courant number is the maximum of the Courant numbers obtained using the phasic velocities.

Generally, in single-phase flow, the Courant number is constrained to be less than one [85] to maintain numerical stability. The step size should satisfy both the local error and Courant number constraints. The local error constraint is first satisfied in Eqn. (4.14) and the step size will be adjusted if the Courant number constraint is not satisfied. The new step size is obtained using:

$$(\Delta t)' = 0.9\Delta t \frac{Co_{max}}{Co}. \tag{4.17}$$

This new step size will replace $(\Delta t)'$ obtained from Eqn. (4.14).

Chapter 5

Phase-Bounded Finite Element Method for Two-Fluid Incompressible Flow Systems

The phase fractions that are used to describe the mixture composition in the two-fluid model are bounded quantities. The numerical solution method used to solve the two-fluid model equations must therefore preserve the boundedness of the phase fractions for accuracy and numerical fidelity. In this chapter, a numerical method for the two-fluid model is developed where inequality constraints are imposed through the implicit nonlinear solver. The method is verified and compared to an existing numerical method.

5.1 Methodology

5.1.1 Simulation Conditions

The two-fluid model is generally well-suited for dilute systems where the phase fraction of the dispersed phase is less than 3% [141]. At higher phase fractions, factors such as turbulence, swarming, *etc.* play an non-negligible role in the hydrodynamical behavior of

the system, resulting in interphase momentum transfer terms that are dependent on the flow regime. In this study, the geometry and physical properties of the system are chosen such that the flow remains dispersed for a long period of time to avoid such dependencies.

The simulation domain used is a two-dimensional channel with gas phase injected from the bottom, as shown in Fig. 5.1. The computational mesh is generated using `GMSH` [142]. The interface drag coefficient C_D is approximated using the Schiller-Naumann drag expression [143]. Momentum transfer due to lift and virtual mass are generally used as "tuning" parameters to increase the agreement between experimental and simulation results [28] and are thus neglected in this work. The Schiller-Naumann drag expression, physical properties and initial and boundary conditions are summarized in Tables 5.1 and 5.2. The inlet gas velocity and phase fraction profiles follow a Gaussian distribution to ensure a smooth transition from the no-slip boundary condition at the walls and to avoid potential issues with discontinuities in the finite element method. In lieu of a wall lubrication force, the phase fraction at the walls is set to zero (liquid wets the wall).

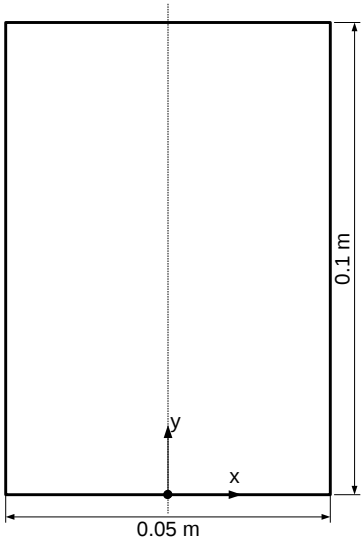


Figure 5.1: Simulation domain.

Table 5.1: Physical properties

Property	Value
Gas density (kg/m^3)	10
Liquid density (kg/m^3)	1000
Gas viscosity (Pa s)	2×10^{-5}
Liquid viscosity (Pa s)	5×10^{-3}
Bubble diameter (m)	10^{-3}
Drag constant	$\max\left[\frac{24}{Re}(1 + 0.15Re^{0.687}), 0.44\right], Re = \frac{\rho_l \|\boldsymbol{v}_r\| d_b}{\mu_l}$

Table 5.2: Initial and boundary conditions.

	Condition
Initial	$\alpha_g(\boldsymbol{x}, 0) = 0$
	$\boldsymbol{v}_g(\boldsymbol{x}, 0) = \boldsymbol{v}_l(\boldsymbol{x}, 0) = \boldsymbol{0}$
	$P(\boldsymbol{x}, 0) = \rho_l g_s(0.1 - y)$
Inlet	$\boldsymbol{v}_g(x, 0, t) = \left(0, \min\left(\frac{t}{t_0}, 1\right) 0.0616 \exp\left[-\frac{\left(\frac{x}{0.025}\right)^2}{2\sigma^2}\right]\right), t_0 = 0.625 \text{ s}, \sigma = 0.1$
	$\boldsymbol{v}_l(x, 0, 0) = \boldsymbol{0}$
	$\alpha_g(x, 0, t) = \min\left(\frac{t}{t_0}, 1\right) 0.026 \exp\left[-\frac{\left(\frac{x}{0.025}\right)^2}{2\sigma^2}\right], t_0 = 0.625 \text{ s}, \sigma = 0.1$
	$\boldsymbol{n} \cdot \boldsymbol{\nabla}(P_l(x, 0, t) - P_l(x, 0, t - \Delta t)) = 0$
Walls	$\boldsymbol{v}_g(\pm 0.025, y, t) = \boldsymbol{v}_l(\pm 0.025, y, t) = \boldsymbol{0}$
	$\alpha_g(\pm 0.025, y, t) = 0$
	$\boldsymbol{n} \cdot \boldsymbol{\nabla}(P_l(\pm 0.025, y, t) - P_l(\pm 0.025, y, t - \Delta t)) = 0$
Outlet	$\boldsymbol{n} \cdot \boldsymbol{\tau}_g(x, 0.1, t) \cdot \boldsymbol{n} = \boldsymbol{n} \cdot \boldsymbol{\tau}_l(x, 0.1, t) \cdot \boldsymbol{n} = 0$
	$\boldsymbol{t} \cdot \boldsymbol{v}_g(x, 0.1, t) = \boldsymbol{t} \cdot \boldsymbol{v}_l(x, 0.1, t) = 0$
	$P_l(x, 0.1, t) = 0$

5.1.2 Numerical Methods

Simulations with the conditions described in the previous section are carried out using the IPCS solver presented in Chapter 4. Adaptive time-stepping is used to constrain the local error to $\leq 10^{-4}$. The second order Heun's method is used to determine the local error of the first order explicit method. It was found in test cases that the largest source of local error is the velocity fields and that the local error between the velocity fields is comparable between tentative and updated velocities. Therefore, only the tentative velocities are used in determining the local error and thus decreasing the number of second order solves to just one per time step.

Equation (2.1) is a pure advection equation, which is susceptible to node-to-node oscillations [144]. To prevent the oscillations from occurring in the simulations, Eqn. (4.6a) is stabilized using the streamline-upwind/Petrov-Galerkin (SUPG) formulation for convection-dominated flows. In the SUPG formulation, the test function in the finite element formulation is modified to allow for upwinding [144]:

$$\varphi' = \varphi + \tau_{SUPG} \boldsymbol{v}_d \cdot \boldsymbol{\nabla} \varphi, \tag{5.1}$$

where φ is the test function and τ_{SUPG} is given by:

$$\tau_{SUPG} = \frac{h}{2\|\boldsymbol{v}_d\|} z, \tag{5.2}$$

$$z = \coth Pe_\alpha - \frac{1}{Pe_\alpha}. \tag{5.3}$$

h is the element length and Pe is the Péclet number. In pure advection transport, the Péclet number is infinite and z is thus equal to one.

In order to maintain phase fraction boundedness, the IPCS scheme is bounded through the use of the nonlinear variational inequality solver SNES [145]. The inequality constraint is formulated as a nonlinear equation that results in a function that is minimized to obtain the constrained solution. The algorithm used to solve this problem is the reduced-space method and is presented in Refs. [146, 147]. An alternative method, a bound-constrained

solver for linear variational inequality from the TAO suite [148], which solves the problem using the trust-region Newton method was also used for comparative purposes. To ensure that the linear variation inequality solver converges at every time step, an additional constraint is placed on the time step such that the time step is decreased until the gradient of the objective function satisfies the specified tolerance. However, this required the time step to consistently be in the order of 10^{-8} while the SNES solver had no such requirement. Thus, the SNES solver is used in the bounded simulations presented in the subsequent sections.

5.2 Results and Discussion

In order to evaluate the effect of phase fraction boundedness on the IPCS scheme, simulations are initially performed with the assumption that the bulk and interfacial pressures are equal, $P_c = P_d = P_{int}$ (Section 2.2). These results are compared to each other and to an alternative finite volume implementation of the two-fluid model in `OpenFOAM`, `twoPhaseEulerFoam`. Following this, simulations are performed for the same conditions, but without the assumption that $P_c = P_{int}$, which is both a more accurate approximation and has been shown to increase the phase fraction interval over which the two-fluid model is well-posed [71, 72, 112].

5.2.1 Effects of Phase Fraction Boundedness, $P_c = P_{int}$

Simulations were performed using physical properties in Table 5.1 and auxiliary conditions in Table 5.2 using the (i) unbounded IPCS solver, (ii) the bounded IPCS solver and (iii) the `twoPhaseEulerFoam` solver from `OpenFOAM`. All of these solvers use the assumption that $P_c = P_{int}$. Figure 5.2 shows α_g and the liquid velocity streamlines at various times for the bounded simulation. Simulation results are shown starting at $t = 1.25\,\text{s}$, when the Rayleigh-Taylor instability [149] first manifests in the formation of a gas phase "plume" as it convects through the liquid phase. As expected, as time increases the plume width increases with increasingly large vortices in the liquid phase velocity forming in its wake.

This morphology has been observed experimentally in the startup period of rectangular bubble columns [150] and its observation serves as qualitative experimental validation of the simulation results. For all three simulations (unbounded IPCS, bounded IPCS, and `twoPhaseEulerFoam`), the bubble plume rises in the column center and vortices in the liquid velocity are observed on each side of the plume.

The average bubble Reynolds numbers at $t = 1.72\,\mathrm{s}$ computed using the definition given in Table 5.1 for the unbounded IPCS, bounded IPCS and `twoPhaseEulerFoam` are 11.542, 11.369 and 11.724, respectively. The results from unbounded IPCS, bounded IPCS and `twoPhaseEulerFoam` solvers at $t = 1.72\,\mathrm{s}$ are given in Fig. 5.3. From Fig. 5.3, both the unbounded and bounded IPCS solvers resulted in similar flow profiles and a qualitatively similar plume is also observed in the `twoPhaseEulerFoam` simulation. However, the gas plume appears to be rising at a faster rate and is wider in the `twoPhaseEulerFoam` simulation than in the IPCS simulations. This difference is possibly due to the difference in how the phase fraction is bounded or in the spatial interpolation scheme.

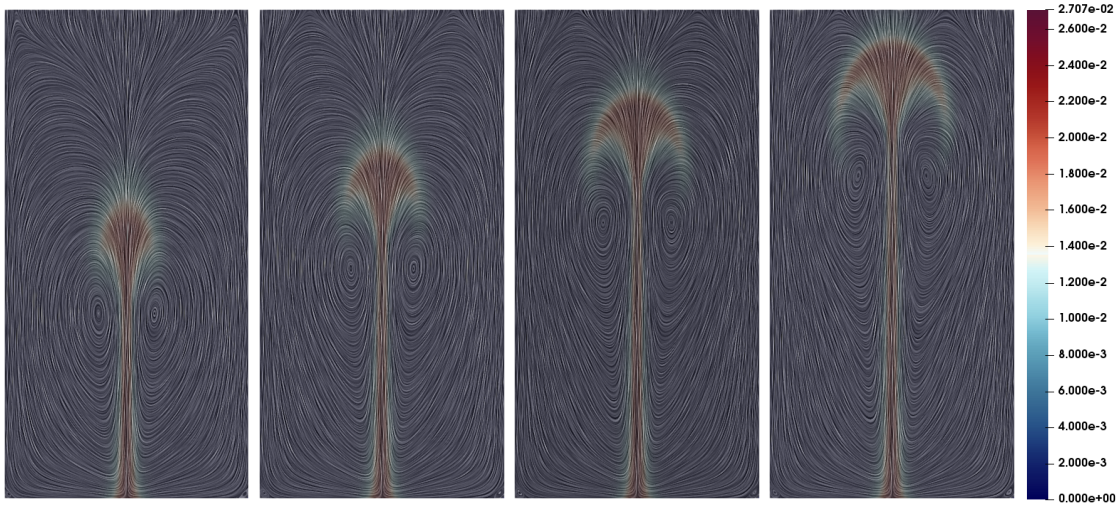


Figure 5.2: Evolution of the phase fraction and liquid velocity streamline over time. Colors denote α_g.

Figure 5.4 shows the evolution of the minimum gas phase fraction, $\min(\alpha_g)$, for the unbounded and bounded IPCS solvers. For the unbounded solver, the magnitude of $\min(\alpha_g)$

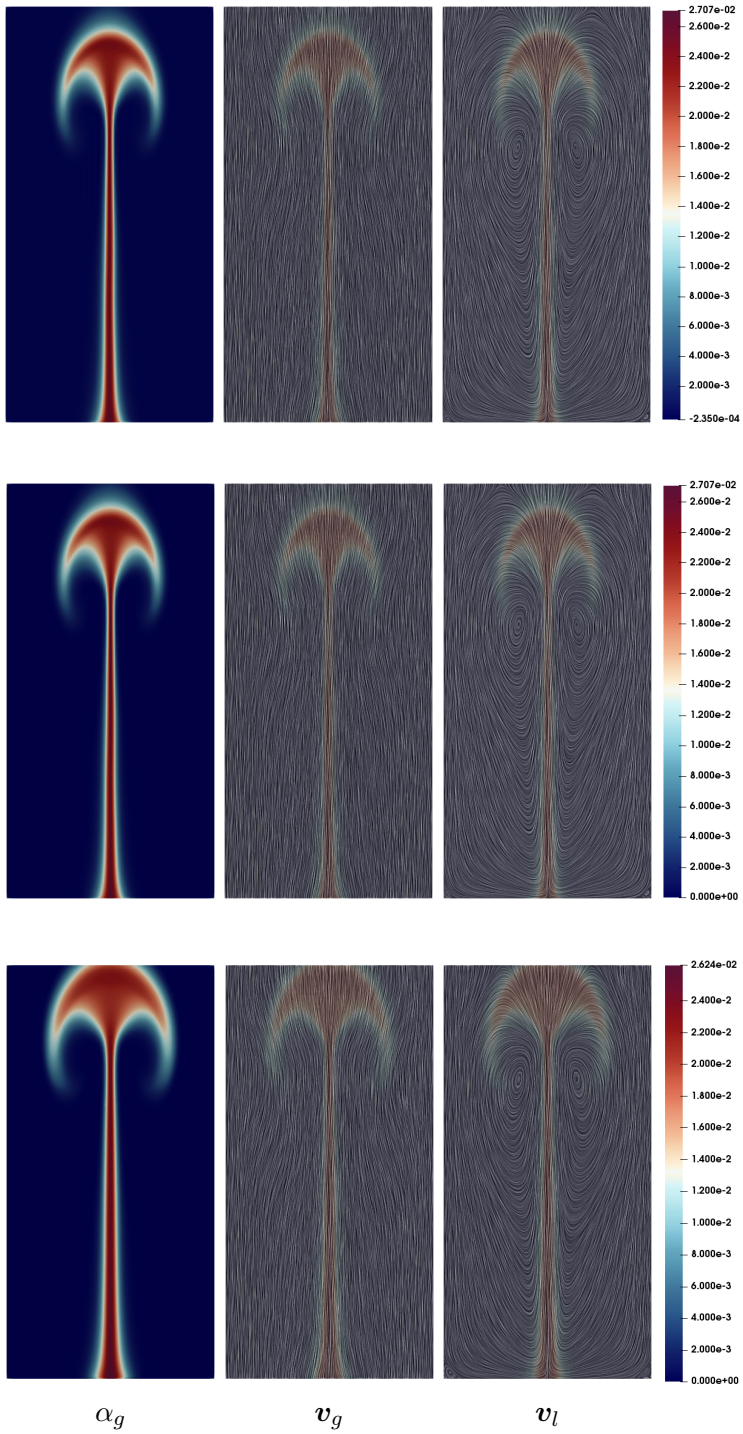


Figure 5.3: Surface plot of (left) phase fraction, (center) gas velocity and (right) liquid velocity at $t = 1.72\,\mathrm{s}$ from (top) unbounded IPCS, (middle) bounded IPCS and (bottom) `twoPhaseEulerFoam`.

is found to be on the order of 10^{-4}, which is on the order of the relative error tolerance of the adaptive time-stepping method, but well above that of the underlying linear solver (10^{-13}). For the bounded solver, the magnitude is within the tolerance of the nonlinear variational solver (10^{-11}).

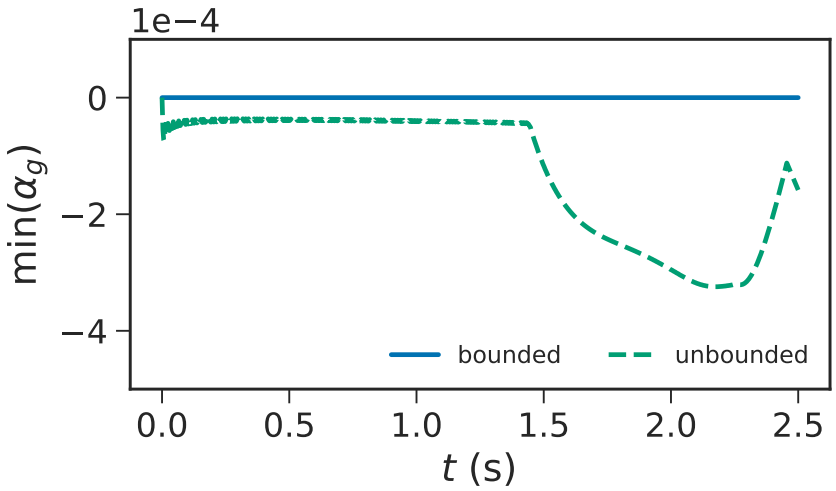


Figure 5.4: Evolution of $\min(\alpha_g)$ over time.

Quantitative comparison of structure of the simulation results from the three solvers was performed using spectral analysis of the phase fraction profiles. Figure 5.5 contains the results from spectral analysis at $t = 1.72\,\mathrm{s}$ for the three different solvers. The histograms of the power spectral density of the phase fraction computed using the IPCS solvers also contained near-zero (in the order of 10^{-40}) power densities that were omitted from Figs. 5.5a and 5.5c. The power spectral density distributions of the IPCS results are similar but drastically different from that of the `twoPhaseEulerFoam` results. This is corroborated by the power spectra shown in Figs. 5.5b, 5.5d and 5.5f. Again, the deviations are possibly due to both the phase fraction different approaches to phase fraction boundedness and/or the spatial interpolation schemes.

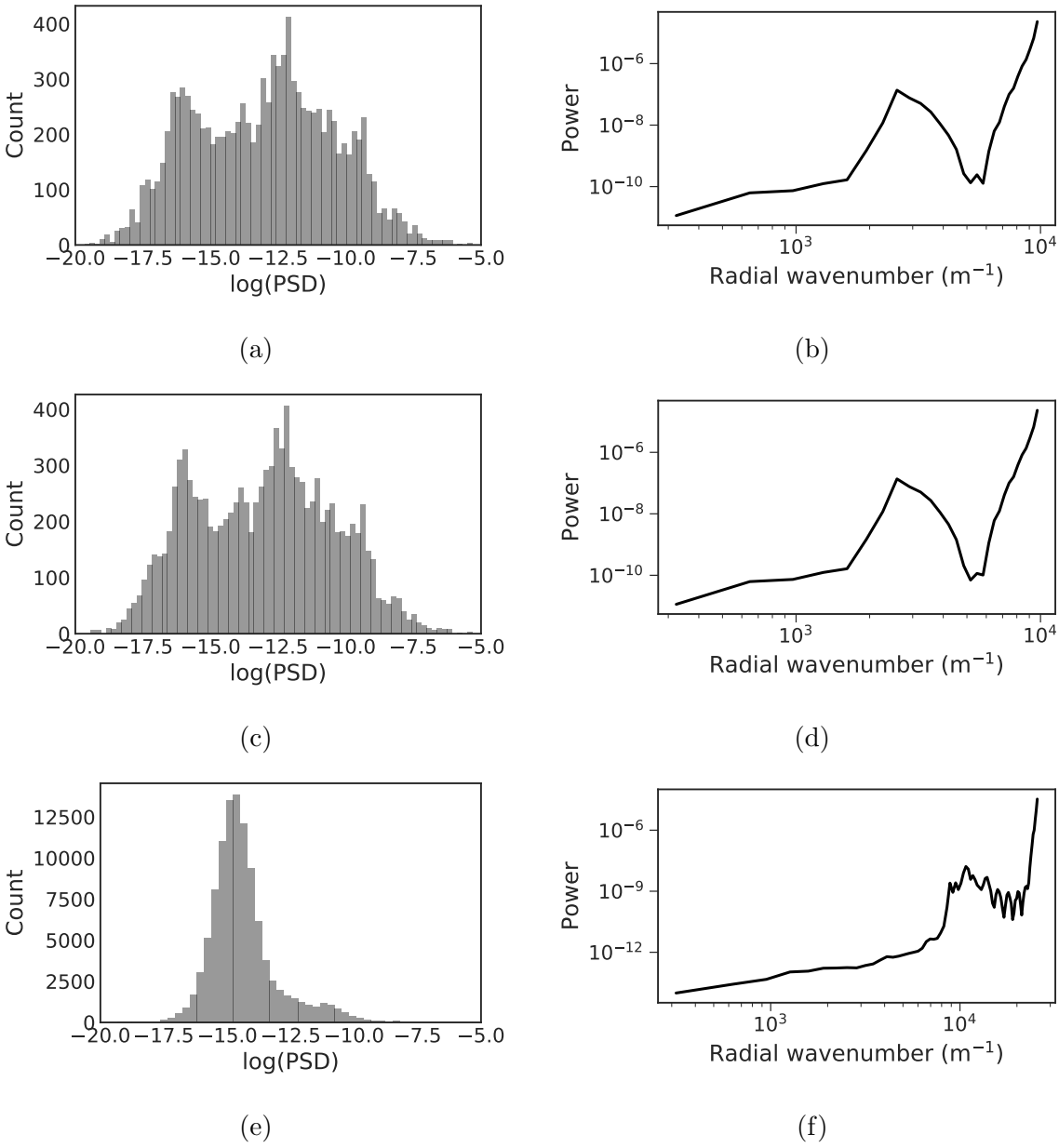


Figure 5.5: Two-dimensional spectral analysis of the phase fraction at $t = 1.72\,\mathrm{s}$ from (top) unbounded IPCS, (middle) bounded IPCS and (bottom) `twoPhaseEulerFoam` solvers. Left: histogram of power spectral density. Right: radially-averaged power spectrum.

5.2.2 Effects of Phase Fraction Boundedness $P_c \neq P_{int}$

The previous simulations were performed with the assumption that the interfacial pressure and the bulk pressure of the continuous phase are equal to each other. In this section, this assumption is removed and a bounded IPCS simulation is performed to assess its effect, which has been shown to increase the phase fraction interval over which the two-fluid model is well-posed [71, 72, 112]. The flow profile at $t = 1.72\,\mathrm{s}$ is shown in Fig. 5.6 and is qualitatively similar to the results in Fig. 5.3. The spatial variation of the gas fractions within the gas phase plume appears to be smoother than the results from the bounded IPCS solver in the previous section (Fig. 5.3). The histogram of the power spectral density (Fig. 5.7a) also shows qualitative agreement with Fig. 5.5c but the radially-averaged power spectra of the two cases are slightly different (Figs. 5.5d and 5.7b).

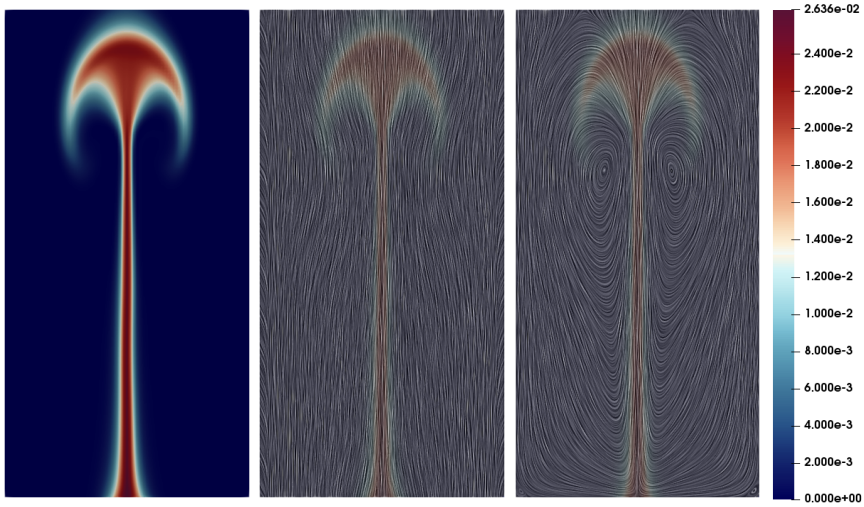
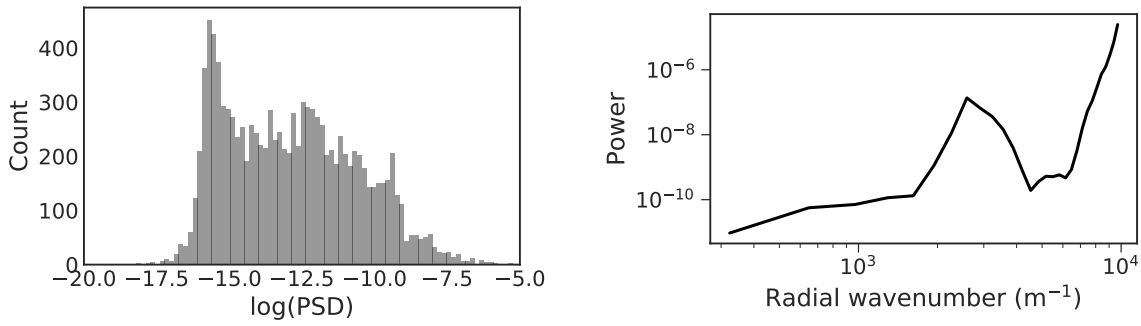


Figure 5.6: Surface plot of (left) phase fraction, (center) gas velocity and (right) liquid velocity at $t = 1.72\,\mathrm{s}$ from bounded IPCS with interfacial pressure.

The time evolution of the overall gas holdup, $\langle \alpha_g \rangle$, from the simulations with the assumption $P_c = P_{int}$ (Section 5.2.1) and the simulation with $P_c \neq P_{int}$ are compared in Fig. 5.8. The overall holdup from the IPCS solvers show little variation from each other, with some expected variation between the solutions with $P_c = P_{int}$ and $P_c \neq P_{int}$. However,



(a) (b)

Figure 5.7: Two-dimensional spectral analysis of the phase fraction at $t = 1.72\,\mathrm{s}$ from bounded IPCS with interfacial pressure. Left: histogram of power spectral density. Right: radially-averaged power spectrum.

the gas holdup obtained from `twoPhaseEulerFoam` is consistently higher, although evolves in a qualitatively similar manner as the IPCS variants. The peak observed in Fig. 5.8 corresponds to the point where the bubble plume is the largest, which is also right before the plume starts to exit the simulation domain. Compared to the IPCS simulations, the time in which the peak occurs is earlier for the `twoPhaseEulerFoam` simulation. This supports the qualitative observations made that the size of the plume and the rate in which the plume moves through the liquid are similar among the IPCS simulations but different when simulations under the same conditions that are performed using `twoPhaseEulerFoam`.

5.3 Conclusions

A phase-bounded numerical method for the two-fluid model is developed using the incremental pressure correction scheme. The phase fraction boundedness is imposed implicitly through the use of the SNES variational inequality solver. Simulations are performed to compare the solution obtained from the phase-bounded method and are found to be similar to the unbounded method, but with the phase fraction equality constraints satisfied within the tolerance of the nonlinear variational inequality solver. The results from the un-

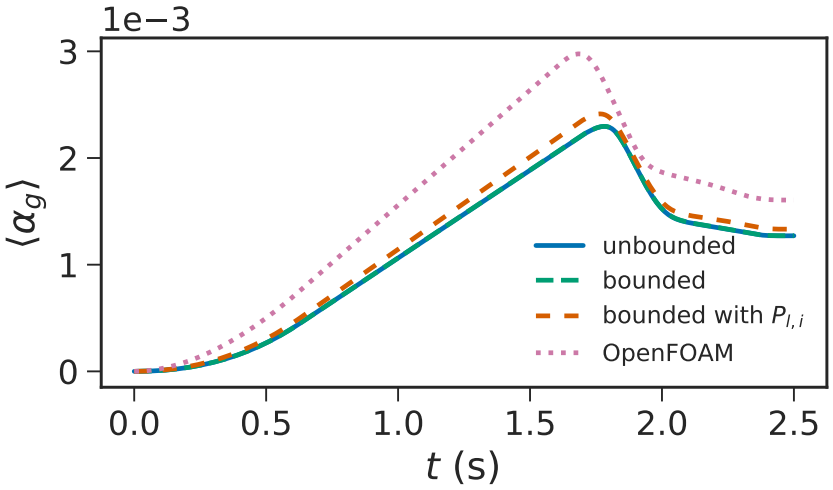


Figure 5.8: Time-evolution of gas holdup.

bounded method exhibit deviations of the minimum value of the gas phase fraction in the domain that are several orders of magnitude greater than the linear solver error tolerance.

Qualitative agreement is found with an alternative bounded two-fluid model solver, the `twoPhaseEulerFoam` solver in the `OpenFOAM` package, although quantitative agreement is not found. This is attributed to either or both the difference in method for imposing phase fraction bounds and approach to spatial interpolation. All numerical solutions are found to agree qualitatively with experimental studies of two-dimensional rectangular bubble columns in the literature.

Finally, the effects of the assumption of the interfacial and bulk pressures of the continuous phase being equal were studied using the phase-bounded method and found to be non-negligible.

Chapter 6

Diffuse-Interface Method for Physical Boundaries

This chapter extends the numerical method developed in Chapter 5 to accommodate a phase-field, ϕ, that describes the diffuse-interface between the fluid mixture and solid boundaries. Simulation conditions in Chapter 5 are replicated with a diffuse-interface defining the solid walls. Simulations of gas-liquid flow past a stationary cylinder are also performed and the simulation results are compared to the results from a conformal mesh. The effects of the interface length-scale and function on the solution are also studied.

6.1 Methodology

The solid physical boundaries are imposed by blending the governing equations of the fluid with the solid Dirichlet boundary conditions. The diffuse-interface is described by the smooth function ϕ, whose value is ± 1 inside the phases and is between $(-1, 1)$ in the interface region [40]:

$$\phi = \begin{cases} -1, & \text{fluid}, \\ 1, & \text{solid}. \end{cases} \tag{6.1}$$

From Eqn. (6.1), the governing equations of the fluid are weighted by $(1-\phi)/2$ to ensure that the equations are active inside the fluid. Similarly, the solid velocity boundary conditions are weighted by $(1+\phi)/2$ so that the conditions are inactive inside the fluid but active in the solid. The gradient of the phase-field is the normal vector from the interface and the Neumann boundary condition can be imposed using $\boldsymbol{n} \approx \boldsymbol{\nabla}\phi/\|\boldsymbol{\nabla}\phi\|$.

An example of this diffuse-interface approach is described using the following Poisson problem:

$$-\nabla^2 y = f \quad \text{on } \Omega, \qquad \boldsymbol{n}\cdot\boldsymbol{\nabla} y = h \quad \text{on } \Gamma_N, \qquad y = g \quad \text{on } \Gamma_D. \tag{6.2}$$

The physical domain is denoted by $\phi = -1$ and the area outside the physical domain by $\phi = 1$. The equation is then weighted by $(1-\phi)/2$ and the Dirichlet condition is weighted by $(1+\phi)/2$:

$$\frac{1-\phi}{2}\big(\nabla^2 y + f\big) + \frac{1+\phi}{2}(y-g) = 0. \tag{6.3}$$

Taking the inner product of Eqn. (6.3) with the test function, φ:

$$\left\langle \frac{1-\phi}{2}\nabla^2 y, \varphi \right\rangle_\Omega + \left\langle \frac{1-\phi}{2} f, \varphi \right\rangle_\Omega + \left\langle \frac{1+\phi}{2}(y-g), \varphi \right\rangle_\Omega = 0. \tag{6.4}$$

The Neumann boundary condition is obtained by applying integration by parts to the Laplacian term:

$$\left\langle \frac{1-\phi}{2}\nabla^2 y, \varphi \right\rangle_\Omega = \left\langle \frac{1-\phi}{2}\boldsymbol{n}\cdot\boldsymbol{\nabla} y, \varphi \right\rangle_{\Gamma'_N} + \left\langle \frac{1}{2}\boldsymbol{\nabla}\phi\cdot\boldsymbol{\nabla} y, \varphi \right\rangle_\Omega - \left\langle \frac{1-\phi}{2}\boldsymbol{\nabla} y, \boldsymbol{\nabla}\varphi \right\rangle_\Omega, \tag{6.5}$$

where Γ'_N is the part of the simulation domain boundary that the Neumann boundary condition applies to and $\boldsymbol{n}$ is the unit normal (outward) of the surface bounding the domain. Substituting this back into Eqn. (6.4) and applying the Neumann boundary condition:

$$\begin{aligned}\left\langle \frac{1-\phi}{2} h, \varphi \right\rangle_{\Gamma'_N} + \left\langle \frac{1}{2} h\|\boldsymbol{\nabla}\phi\|, \varphi \right\rangle_\Omega - \left\langle \frac{1-\phi}{2}\boldsymbol{\nabla} y, \boldsymbol{\nabla}\varphi \right\rangle_\Omega \\ + \left\langle \frac{1-\phi}{2} f, \varphi \right\rangle_\Omega + \left\langle \frac{1+\phi}{2}(y-g), \varphi \right\rangle_\Omega = 0.\end{aligned} \tag{6.6}$$

Equation (6.6) is the weak formulation of Eqn. (6.2) with a diffuse-interface. The Neumann boundary condition is imposed via the first and/or second terms, depending on whether Γ'_N exists or not. Similarly, should the Dirichlet boundary condition also apply to parts of the simulation domain boundary, the boundary condition is applied by setting $y = g$ at Γ'_D.

6.1.1 Numerical Method

Prior to delving into the diffuse-interface method, the notation that will be used for the time discretization must be defined. The governing equations are solved using an adaptive second/third order semi-implicit Adams-Bashforth/Backward-Differentiation (AB/BDI23) scheme [151]. The third order AB/BDI3 scheme is used to estimate the local error of the second order scheme. The explicit terms in the equation are discretized using the Adams-Bashforth scheme and the time derivative is discretized using backward-differentiation [151]. The following notation will be used to denote the numerator of the discretized time derivative:

$$\boldsymbol{v}^{*'} = a_0 \boldsymbol{v}^* + \sum_{j=1}^{k} a_j \boldsymbol{v}^{n+1-j}, \tag{6.7}$$

$$\alpha^{(n+1)'} = \sum_{j=0}^{k} a_j \alpha^{n+1-j}, \tag{6.8}$$

where a_j is a coefficient associated with backward-differentiation that will later be defined and k is the order of the method. The discretized explicit terms will be denoted as follows:

$$f^{n'} = \sum_{j=0}^{k-1} b_j f^{n-j}, \tag{6.9}$$

where b_j is a coefficient associated with the Adams-Bashforth scheme.

To account for the variable step size, the coefficients a_j and b_j are functions of the previous time steps. This has the benefit of not having to interpolate between time steps

to obtain the solution at $t^n - \Delta t$ and $t^n - 2\Delta t$. Let r be the ratio of the previous step size to the current step size:

$$r^n = \frac{t^n - t^{n-1}}{\Delta t}, \tag{6.10}$$

where $t^n - t^{n-1}$ is the step size used in the previous time step. The coefficients in AB/BDI2 can then be written in terms of r^n [151]:

$$a_0 = \frac{2 + r^n}{1 + r^n}, \qquad a_1 = -1 - \frac{1}{r^n}, \qquad a_2 = \frac{1}{r^n(1 + r^n)}, \tag{6.11}$$

$$b_0 = -a_1, \qquad b_1 = 1 + a_1. \tag{6.12}$$

The same can be done for AB/BDI3 where $r^{n-1} = (t^{n-1} - t^{n-2})/\Delta t$ [151]:

$$\begin{aligned}
a_0 &= 1 + \frac{1}{1 + r^n} + \frac{1}{1 + r^n + r^{n-1}}, \\
a_1 &= -\frac{(1 + r^n)(1 + r^n + r^{n-1})}{r^n(r^n + r^{n-1})}, \\
a_2 &= \frac{1 + r^n + r^{n-1}}{r^n r^{n-1}(1 + r^n)}, \\
a_3 &= -\frac{1 + r^n}{r^{n-1}(r^n + r^{n-1})(1 + r^n + r^{n-1})},
\end{aligned} \tag{6.13}$$

$$\begin{aligned}
b_0 &= -a_1, \\
b_1 &= -a_2(1 + r^n), \\
b_2 &= -a_3(1 + r^n + r^{n-1}).
\end{aligned} \tag{6.14}$$

The step size is chosen using the adaptive time-stepping scheme outlined in Section 4.3.

6.1.2 Diffuse-Interface for Two-Fluid Model Equations

The diffuse solid-fluid interface is imposed by blending the governing equations of the two-fluid model (Eqn. (4.3)) and the solid Dirichlet boundary condition together. This is achieved by weighting the governing equations and solid boundary condition by $(1 - \phi)/2$ and $(1 + \phi)/2$, respectively. The weighting allows for integrals over the physical domain

to be reformulated into volume integrals over the simulation domain [56]. The resulting system of equations is as follows:

$$\frac{1-\phi}{2}\left(\frac{\tilde{\boldsymbol{v}}_q^{*'}}{\Delta t}+\tilde{\boldsymbol{v}}_q^{n'}\cdot\tilde{\boldsymbol{\nabla}}\tilde{\boldsymbol{v}}_q^{n'}\right)=\frac{1-\phi}{2}\left[\boldsymbol{RHS}_q^n+\frac{1}{Re_q}\frac{\tilde{\boldsymbol{\nabla}}\alpha_q^{n'}\cdot\tilde{\boldsymbol{\tau}}_q^*}{\alpha_q^{n'}}+\frac{1}{Re_q}\tilde{\boldsymbol{\nabla}}\cdot\tilde{\boldsymbol{\tau}}_q^*+\frac{1}{Fr^2}\tilde{\boldsymbol{g}}\right] \quad \text{in} \quad \Omega, \tag{6.15}$$

where:

$$\boldsymbol{RHS}_l^n=-Eu_l\tilde{\boldsymbol{\nabla}}\tilde{P}_l^n+\frac{3}{4}\frac{\alpha_g^{n'}}{\alpha_l^{n'}}\frac{C_D}{\tilde{d}_b}\left\|\tilde{\boldsymbol{v}}_r^{n'}\right\|\tilde{\boldsymbol{v}}_r^{n'}-C_P\tilde{\boldsymbol{v}}_r^{n'}\cdot\tilde{\boldsymbol{v}}_r^{n'}\frac{\tilde{\boldsymbol{\nabla}}\alpha_l^{n'}}{\alpha_l^{n'}}, \tag{6.16a}$$

$$\boldsymbol{RHS}_g^n=-Eu_g\tilde{\boldsymbol{\nabla}}\tilde{P}_l^n+C_P\tilde{\boldsymbol{\nabla}}\left(\tilde{\boldsymbol{v}}_r^{n'}\cdot\tilde{\boldsymbol{v}}_r^{n'}\right)\frac{\rho_l}{\rho_g}-\frac{3}{4}\frac{\rho_l}{\rho_g}\frac{C_D}{\tilde{d}_b}\left\|\tilde{\boldsymbol{v}}_r^{n'}\right\|\tilde{\boldsymbol{v}}_r^{n'}, \tag{6.16b}$$

with the following boundary conditions:

$$\frac{1+\phi}{2}\tilde{\boldsymbol{v}}_q^*=\boldsymbol{0} \quad \text{in} \quad \Omega, \tag{6.17a}$$

$$\frac{1-\phi}{2}\tilde{\boldsymbol{v}}_q^*=\frac{1-\phi}{2}\tilde{\boldsymbol{v}}_{q,BC}^{n+1} \quad \text{on} \quad \Gamma'_D, \tag{6.17b}$$

$$\boldsymbol{n}\cdot\frac{1-\phi}{2}\tilde{\boldsymbol{\tau}}_q^*=\boldsymbol{0} \quad \text{on} \quad \Gamma'_N. \tag{6.17c}$$

The weak formulation of Eqn. (6.15) follows the same procedure outlined in Section 4.2 but with the terms weighted by $(1-\phi)/2$. The differences are in the treatment of the viscous stress term that results in the Neumann boundary condition and the addition of solid Dirichlet velocity boundary conditions. Taking the inner product of the viscous stress term:

$$\frac{1-\phi}{2}\frac{1}{Re_q}\tilde{\boldsymbol{\nabla}}\cdot\tilde{\boldsymbol{\tau}}_q^*,$$

with the test function and using integration by parts results in the following:

$$\begin{aligned}\left\langle \frac{1-\phi}{2}\frac{1}{Re_q}\tilde{\boldsymbol{\nabla}}\cdot\tilde{\boldsymbol{\tau}}_q^*, \boldsymbol{\varphi}_q \right\rangle_\Omega =& \left\langle \frac{1-\phi}{2}\frac{1}{Re_q}\boldsymbol{n}\cdot\tilde{\boldsymbol{\tau}}_q^*, \boldsymbol{\varphi}_q \right\rangle_{\Gamma'_N} + \left\langle \frac{1}{2Re_q}\tilde{\boldsymbol{\nabla}}\phi\cdot\tilde{\boldsymbol{\tau}}_q^*, \boldsymbol{\varphi}_q \right\rangle_\Omega \\ &- \left\langle \frac{1-\phi}{2}\frac{1}{Re_q}\tilde{\boldsymbol{\tau}}_q^*, \tilde{\boldsymbol{\nabla}}\boldsymbol{\varphi}_q \right\rangle_\Omega .\end{aligned} \tag{6.18}$$

The second term in the right-hand side of Eqn. (6.18) allows for the imposition of a Neumann boundary condition at the solid-fluid interface. In this work, the boundary condition at the solid-fluid interface is a Dirichlet boundary condition and the term is therefore left unconstrained. The weak formulation is thus:

$$\begin{aligned}&\left\langle \frac{1-\phi}{2}\frac{\tilde{\boldsymbol{v}}_q^{*'}}{\Delta t}, \boldsymbol{\varphi}_q \right\rangle_\Omega + \left\langle \frac{1+\phi}{2}\frac{a_0\tilde{\boldsymbol{v}}_q^*}{\Delta t}, \boldsymbol{\varphi}_q \right\rangle_\Omega \\ &\qquad = -\left\langle \frac{1-\phi}{2}\tilde{\boldsymbol{v}}_q^{n'}\cdot\tilde{\boldsymbol{\nabla}}\tilde{\boldsymbol{v}}_q^{n'}, \boldsymbol{\varphi}_q \right\rangle_\Omega + \left\langle \frac{1-\phi}{2}\boldsymbol{RHS}_q^n, \boldsymbol{\varphi}_q \right\rangle_\Omega \\ &\qquad + \left\langle \frac{1-\phi}{2}\frac{1}{Re_q}\frac{\tilde{\boldsymbol{\nabla}}\alpha_q^{n'}\cdot\tilde{\boldsymbol{\tau}}_q^*}{\alpha_q^{n'}}, \boldsymbol{\varphi}_q \right\rangle_\Omega + \left\langle \frac{1-\phi}{2}\frac{1}{Re_q}\boldsymbol{n}\cdot\tilde{\boldsymbol{\tau}}_q^*, \boldsymbol{\varphi}_q \right\rangle_{\Gamma'_N} \\ &\qquad - \left\langle \frac{1-\phi}{2}\frac{1}{Re_q}\tilde{\boldsymbol{\tau}}_q^*, \tilde{\boldsymbol{\nabla}}\boldsymbol{\varphi}_q \right\rangle_\Omega + \left\langle \frac{1-\phi}{2}\frac{1}{Fr^2}\tilde{\boldsymbol{g}}, \boldsymbol{\varphi}_q \right\rangle_\Omega ,\end{aligned} \tag{6.19}$$

where the solid boundary condition is weighted by $a_0/\Delta t$ for consistency. The pressure Poisson equation is derived from Eqn. (6.15) by taking the difference between the weighted momentum equation for $\tilde{\boldsymbol{v}}_q^{n+1}$ and $\tilde{\boldsymbol{v}}_q^*$ and neglecting the contributions of convection, viscous stress and interphase momentum transfer:

$$-\tilde{\boldsymbol{\nabla}}\cdot\left[\frac{1-\phi}{2}\sum_q Eu_q\alpha_q^{n'}\tilde{\boldsymbol{\nabla}}\left(\tilde{P}_l^{n+1}-\tilde{P}_l^n\right)\right] = \frac{a_0}{\Delta t}\tilde{\boldsymbol{\nabla}}\cdot\left[\frac{1-\phi}{2}\sum_q\left(\alpha_q^{n+1}\tilde{\boldsymbol{v}}_q^{n+1}-\alpha_q^n\tilde{\boldsymbol{v}}_q^*\right)\right]. \tag{6.20}$$

The right-hand side term can be separated into two terms:

$$\begin{aligned}\frac{a_0}{\Delta t}\tilde{\boldsymbol{\nabla}}\cdot\left[\frac{1-\phi}{2}\sum_q\left(\alpha_q^{n+1}\tilde{\boldsymbol{v}}_q^{n+1}-\alpha_q^n\tilde{\boldsymbol{v}}_q^*\right)\right] =& -\frac{a_0}{2\Delta t}\tilde{\boldsymbol{\nabla}}\phi\cdot\sum_q\left(\alpha_q^{n+1}\tilde{\boldsymbol{v}}_q^{n+1}-\alpha_q^n\tilde{\boldsymbol{v}}_q^*\right)\\ &+\frac{a_0}{\Delta t}\frac{1-\phi}{2}\tilde{\boldsymbol{\nabla}}\cdot\sum_q\left(\alpha_q^{n+1}\tilde{\boldsymbol{v}}_q^{n+1}-\alpha_q^n\tilde{\boldsymbol{v}}_q^*\right).\end{aligned} \tag{6.21}$$

The first term is only active at the solid-fluid interface and given that the phase fraction and velocity of the solid are always known, this term is assumed to be negligible. Using the incompressibility condition, the pressure Poisson equation for two-phase flow using the diffuse-interface method is thus:

$$\tilde{\boldsymbol{\nabla}}\cdot\left[\frac{1-\phi}{2}\sum_q Eu_q\alpha_q^{n'}\tilde{\boldsymbol{\nabla}}\left(\tilde{P}_l^{n+1}-\tilde{P}_l^n\right)\right]=\frac{a_0}{\Delta t}\frac{1-\phi}{2}\tilde{\boldsymbol{\nabla}}\cdot\left(\sum_q\ \alpha_q^n\tilde{\boldsymbol{v}}_q^*\right), \tag{6.22}$$

with the following weak formulation:

$$\begin{aligned}\left\langle\frac{1-\phi}{2}\sum_q Eu_q\alpha_q^{n'}\boldsymbol{n}\cdot\tilde{\boldsymbol{\nabla}}\left(\tilde{P}_l^{n+1}-\tilde{P}_l^n\right),\varphi_p\right\rangle_{\Gamma'_D} &-\left\langle\frac{1-\phi}{2}\sum_q Eu_q\alpha_q^{n'}\tilde{\boldsymbol{\nabla}}\left(\tilde{P}_l^{n+1}-\tilde{P}_l^n\right),\tilde{\varphi}_p\right\rangle_\Omega\\ &=\left\langle\frac{a_0}{\Delta t}\frac{1-\phi}{2}\tilde{\boldsymbol{\nabla}}\cdot\left(\sum_q\alpha_q^{n'}\tilde{\boldsymbol{v}}_q^*\right),\varphi_p\right\rangle_\Omega.\end{aligned} \tag{6.23}$$

The new velocity update equation is simply sum of the update equation from IPCS weighted by $(1-\phi)/2$ and the solid Dirichlet boundary condition weighted by $(1+\phi)/2$:

$$\left\langle\frac{1-\phi}{2}a_0\frac{\tilde{\boldsymbol{v}}_q^{n+1}-\tilde{\boldsymbol{v}}_q^*}{\Delta t},\boldsymbol{\varphi}_q\right\rangle_\Omega+\left\langle\frac{1+\phi}{2}\frac{a_0\tilde{\boldsymbol{v}}_q^{n+1}}{\Delta t},\boldsymbol{\varphi}_q\right\rangle_\Omega=-\left\langle\frac{1-\phi}{2}Eu_q\tilde{\boldsymbol{\nabla}}\left(\tilde{P}_l^{n+1}-\tilde{P}_l^n\right),\boldsymbol{\varphi}_q\right\rangle_\Omega. \tag{6.24}$$

The boundary condition for the gas fraction, α_g, at the solid-fluid interface is $\alpha_g=0$ (liquid wets the wall). Using the same blending procedure to apply the boundary condition yields the following:

$$\left\langle \frac{1-\phi}{2}\frac{\alpha_g^{(n+1)'}}{\Delta t}, \varphi_\alpha \right\rangle_\Omega + \left\langle \frac{1+\phi}{2}\frac{a_0\alpha_g^{n+1}}{\Delta t}, \varphi_\alpha \right\rangle_\Omega + \left\langle \frac{1-\phi}{2}\tilde{\boldsymbol{\nabla}}\cdot\left(\alpha_g^{n+1}\tilde{\boldsymbol{v}}_g^{n+1}\right), \varphi_\alpha \right\rangle_\Omega = 0. \quad (6.25)$$

6.1.3 Simulation Conditions

The diffuse-interface method is used to impose boundary conditions in dispersed gas-liquid simulations of a two-dimensional channel (Fig. 6.1) and flow past a stationary cylinder (Fig. 6.2). The physical properties of the fluids are the same as those reported in Table 5.1. The width of the channel in Fig. 6.1 is twice that of the simulation domain in Chapter 5. The channel walls will be imposed using a phase-field and the remaining boundary conditions are the same as in Chapter 5. The new inlet boundary conditions are given in Table 6.1. For the case of flow past a cylinder, parabolic velocity and gas fraction profiles are used at the inlet (Table 6.2), no-slip and zero gas fraction conditions are imposed at the channel and cylinder walls and outflow conditions are used at the outlet.

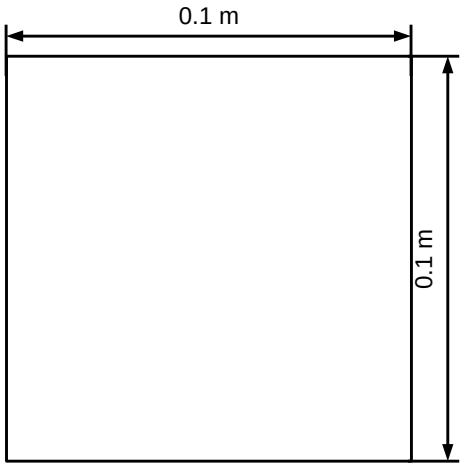


Figure 6.1: Simulation domain for gas-liquid flow inside a channel with the diffuse-interface method.

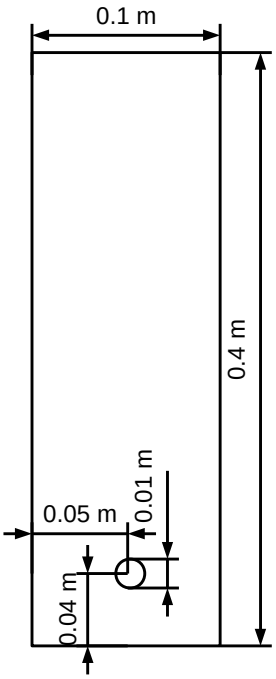


Figure 6.2: Simulation domain for gas-liquid flow past a stationary cylinder with the diffuse-interface method.

Table 5.1: Physical properties (repeated from page 55)

Property	Value
Gas density (kg/m^3)	10
Liquid density (kg/m^3)	1000
Gas viscosity (Pa s)	2×10^{-5}
Liquid viscosity (Pa s)	5×10^{-3}
Bubble diameter (m)	10^{-3}
Drag constant	$\max\left[\frac{24}{Re}(1+0.15Re^{0.687}), 0.44\right], Re = \frac{\rho_l \|\boldsymbol{v}_r\| d_b}{\mu_l}$

Table 6.1: Initial and inlet conditions for gas-liquid channel flow with diffuse-interface.

	Condition
Initial	$\alpha_g(\boldsymbol{x}, 0) = 0$
	$\boldsymbol{v}_g(\boldsymbol{x}, 0) = \boldsymbol{v}_l(\boldsymbol{x}, 0) = \boldsymbol{0}$
	$P(\boldsymbol{x}, 0) = \rho_l g_s(0.1 - y)$
Inlet	$\boldsymbol{v}_g(x, 0, t) = \left(0, \min\left(\frac{t}{t_0}, 1\right)\frac{1-\phi}{2} 0.0616 \exp\left[-\frac{\left(\frac{x}{0.025}\right)^2}{2\sigma^2}\right]\right), t_0 = 0.625 \text{ s}, \sigma = 0.1$
	$\boldsymbol{v}_l(x, 0, 0) = \boldsymbol{0}$
	$\alpha_g(x, 0, t) = \min\left(\frac{t}{t_0}, 1\right)\frac{1-\phi}{2} 0.026 \exp\left[-\frac{\left(\frac{x}{0.025}\right)^2}{2\sigma^2}\right], t_0 = 0.625 \text{ s}, \sigma = 0.1$
	$\boldsymbol{n} \cdot \frac{1-\phi}{2} \boldsymbol{\nabla}(P_l(x, 0, t) - P_l(x, 0, t - \Delta t)) = 0$

Table 6.2: Initial and inlet conditions for gas-liquid flow past a cylinder.

	Condition
Initial	$\alpha_g(\boldsymbol{x}, 0) = 0$
	$\boldsymbol{v}_g(\boldsymbol{x}, 0) = \boldsymbol{v}_l(\boldsymbol{x}, 0) = \boldsymbol{0}$
	$P(\boldsymbol{x}, 0) = \rho_l g_s(0.4 - y)$
Inlet	$\boldsymbol{v}_g(x, 0, t) = \left(0, \min\left(\frac{t}{t_0}, 1\right) 0.0616(0.025 - x^2)\right), t_0 = 0.625 \text{ s}$
	$\boldsymbol{v}_l(x, 0, 0) = \boldsymbol{0}$
	$\alpha_g(x, 0, t) = \min\left(\frac{t}{t_0}, 1\right) 0.02(0.025 - x^2), t_0 = 0.625 \text{ s}$
	$\boldsymbol{n} \cdot \boldsymbol{\nabla} P(x, 0, t) = 0$

6.2 Results and Discussion

To validate the diffuse-interface method for imposing static solid boundaries, simulations of two-phase flow using the diffuse-interface are compared to simulation results from a boundary-conformal mesh for both channel flow and flow past a cylinder. The effect of the diffuse-interface length-scale and function type on the solution and the performance of the method are discussed.

6.2.1 Channel Flow

The phase-field that defines the channel is described using the following hyperbolic tangent function:

$$\phi(\tilde{\boldsymbol{x}}) = \tanh\left(\frac{|\tilde{x}| - \tilde{x}_c}{0.5\epsilon}\right), \tag{6.26}$$

where $\tilde{x}_c = 0.5$ is the scaled distance from the centerline to the channel wall and ϵ is a parameter associated with the width of the diffuse-interface. The function will asymptotically approach $\phi = -1$ and $\phi = 1$, ensuring a smooth transition between the phases. The scaled width of the interface, η, is approximated by the distance between $\phi = -0.999$ and $\phi = 0.999$ which is given by $\eta = \epsilon \tanh^{-1}(0.999)$.

The presence of the diffuse-interface alters the way the no-slip boundary condition is imposed at the channel walls. In the case of a boundary-conformal mesh, the velocities at the walls are set to zero. However, in the diffuse-interface method, the no-slip condition is blended with the governing equations for the two-fluid model. The sharpness of the velocity gradient from the channel walls to the bulk is now a function of the diffuse-interface function, interface width and the discretization scheme.

In this study, the spatial discretization scheme is the same for all of the simulations but two different orders of the temporal discretization scheme are used to estimate the local error. The difference between near-wall velocity gradients from second and third order AB/BDI methods are the largest contributor to the local error and a local error tolerance of $\epsilon_l = 10^{-4}$ resulted in very small step sizes. This issue is particularly significant in cases where the diffuse-interface is large such as in channel flow. To alleviate the constraint on

the step size, only the local error inside the fluid domain, where $\phi \leq -0.999$, is considered when computing the new step size and the local error tolerance is relaxed to $\epsilon_l = 10^{-3}$.

The gas phase fraction profile at $t = 1.72\,\mathrm{s}$ obtained from a simulation with a diffuse-interface given by Eqn. (6.26) and $\epsilon = 0.02$ is shown in Fig. 6.3. The profile for ϕ is superimposed onto the image and thresholded to only show $\phi \geq -0.999$. Qualitatively, the phase fraction profile is in agreement to that observed in Chapter 5. Figure 6.4 shows the gas and liquid velocity streamlines inside the box given by $x \in [-0.025, 0.025]$ and $y \in [0, 0.1]$ at the same time step. The reference solution from Chapter 5 (Fig. 5.6) is reproduced in this chapter for comparative purposes. From Figs. 5.6 and 6.4, the velocity profiles of both gas and liquid phases are similar with liquid recirculating in the wake of the bubble plume.

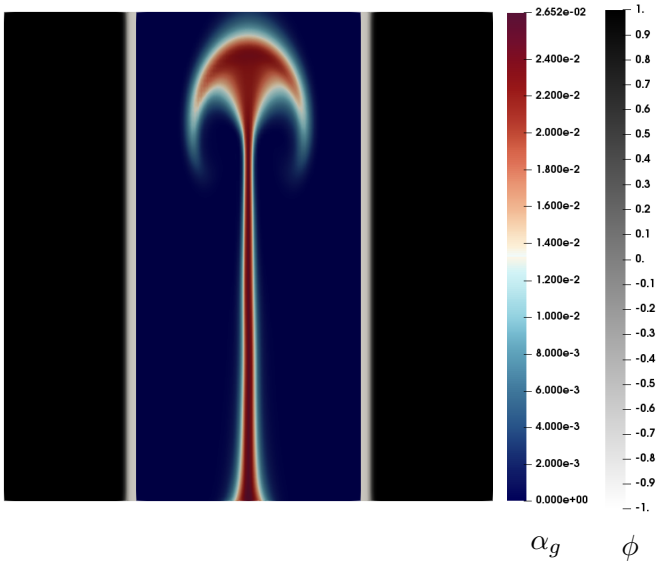


Figure 6.3: Surface plot of α_g at $t = 1.72\,\mathrm{s}$ with hyperbolic tangent diffuse-interface and $\epsilon = 0.02$. The grayscale colorbar denotes the phase-field that describes the diffuse-interface, thresholded to show $\phi \geq -0.999$.

In addition to qualitative comparisons of the phase fraction profile and velocity streamlines, the time evolution of the gas hold-up from the diffuse-interface simulation will also be compared to that of the reference solution from Chapter 5. The gas hold-up in the

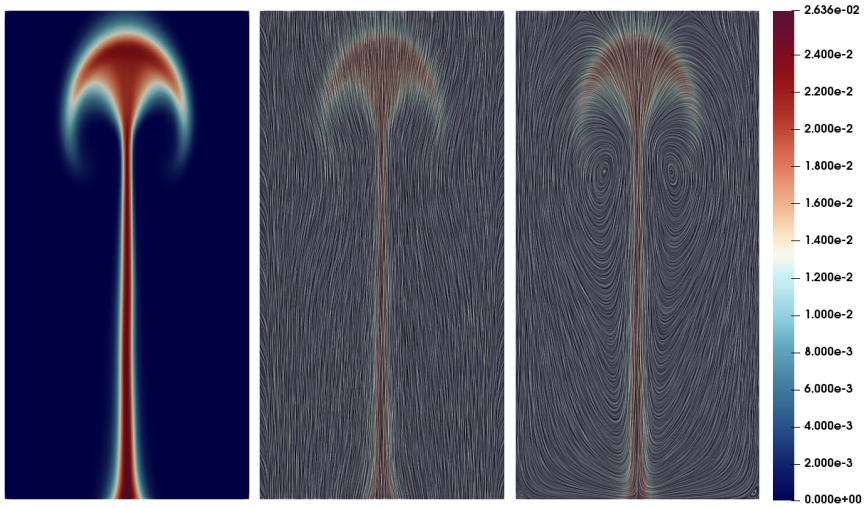


Figure 5.6: Surface plot of (left) phase fraction, (center) gas velocity and (right) liquid velocity at $t = 1.72\,\mathrm{s}$ from bounded IPCS with interfacial pressure. (repeated from page 62)

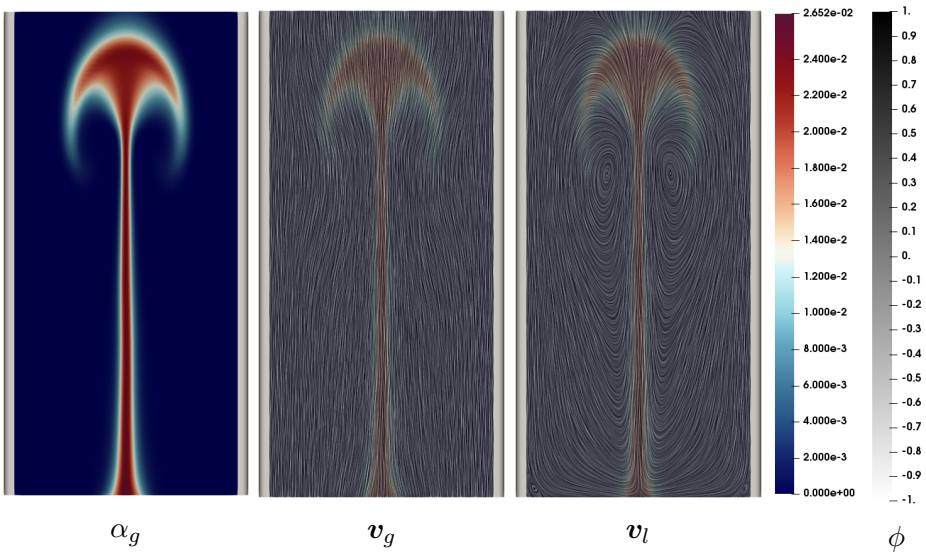


Figure 6.4: Surface plot of (left) α_g, (center) gas velocity and (right) liquid velocity at $t = 1.72\,\mathrm{s}$ with hyperbolic tangent diffuse-interface and $\epsilon = 0.02$.

diffuse-interface simulation is computed as follows:

$$\langle \alpha_g \rangle = \frac{\int_\Omega \frac{1-\phi}{2} \alpha_g \mathrm{d}\Omega}{\int_\Omega \frac{1-\phi}{2} \mathrm{d}\Omega}, \tag{6.27}$$

where denominator is the volume of the physical domain. This comparison is reported in the subsequent sections.

Effect of Interface Length-Scale

As previously discussed, the nature of the diffuse-interface can affect simulation results and how well they closely replicate the boundary-conformal mesh solutions. This section will explore the effect of interface length-scale on the solution. Simulations of the same channel flow system is repeated with $\epsilon = 0.01$, 0.04, 0.08 and 0.1. Figure 6.5 shows how the $\phi = \tanh(x/0.5\epsilon)$ profile changes with different values of ϵ. $\epsilon = 0.01$ corresponds to the case where the interface is sharper and $\epsilon = 0.1$ to the case where the interface is very diffuse.

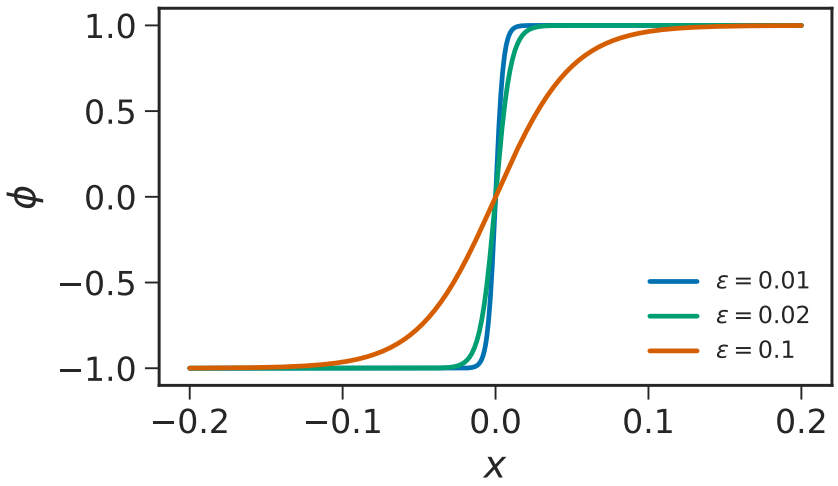


Figure 6.5: Comparison of diffuse-interface width generated using the same hyperbolic tangent function with varying ϵ.

As the interface widens, the contribution of local error from blending increases. But given that this contribution is localized to the interface and that only the local error inside the fluid is considered when computing the new step size, the step size is comparable between all values of ϵ. Figures 6.6 and 6.7 show the gas phase fraction at $t = 1.72\,\mathrm{s}$ for simulations with $\epsilon = 0.01$ and $\epsilon = 0.1$, respectively. Visually, gas fraction profile from $\epsilon = 0.01$ is nearly identical to the case with $\epsilon = 0.02$ but the profile from $\epsilon = 0.1$ is notably different from $\epsilon = 0.02$. In Fig. 6.7, noticeable "wobbling" is observed in the gas column below the plume and the plume is much narrower. This is due to the interface being very diffuse and the effect of the solid boundary conditions is smeared further into the fluid domain.

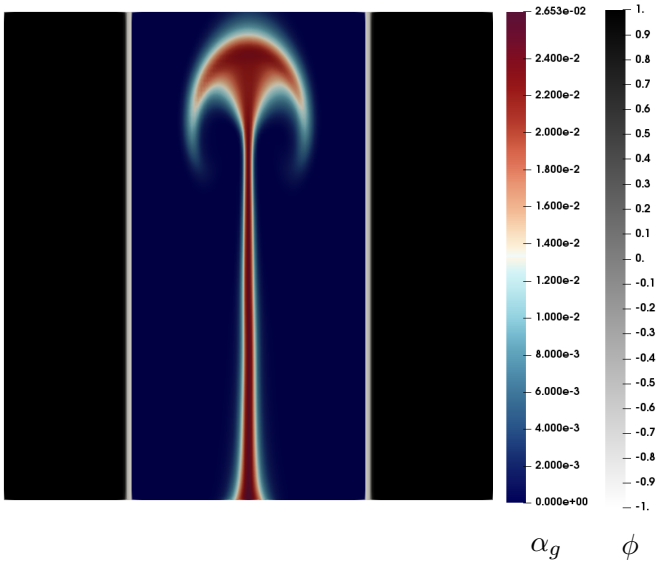


Figure 6.6: Surface plot of α_g at $t = 1.72\,\mathrm{s}$ with hyperbolic tangent diffuse-interface and $\epsilon = 0.01$. The grayscale colorbar denotes the phase-field that describes the diffuse-interface, thresholded to show $\phi \geq -0.999$.

The gas and liquid velocity streamlines from $\epsilon = 0.01$ and $\epsilon = 0.1$ are shown in Figs. 6.8 and 6.9, respectively. The streamlines from $\epsilon = 0.01$ are qualitatively similar to those observed in Figs. 5.6 and 6.4. However, the streamlines from $\epsilon = 0.1$ are different from the other simulations. The gas velocity streamlines appear to exhibit less curvature in the wake of the bubble plume and the liquid velocity vortices in the wake of the plume are

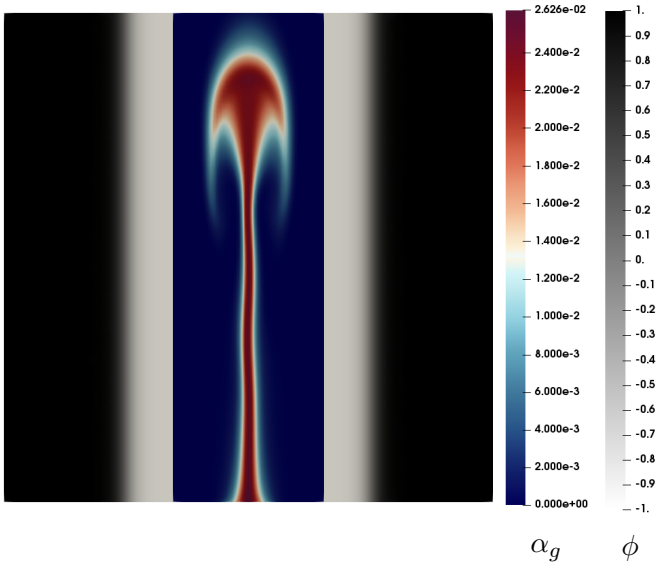


Figure 6.7: Surface plot of α_g at $t = 1.72\,\mathrm{s}$ with hyperbolic tangent diffuse-interface and $\epsilon = 0.1$. The grayscale colorbar denotes the phase-field that describes the diffuse-interface, thresholded to show $\phi \geq -0.999$.

narrower due to the highly diffuse nature of the interface.

Figure 6.10 shows the time evolution of the overall gas hold-up, $\langle \alpha_g \rangle$, inside the channel up to 2.5 s from the hyperbolic tangent diffuse-interface simulations and the reference solution. At narrow interface widths, evolution of the gas hold-up follows the same pattern as the reference solution and the magnitude of overall hold-up is almost identical. However, for $\epsilon = 0.1$, the evolution of the hold-up is similar to the reference solution only up to the point where the bubble plume leaves the channel. After this point, the hold-up deviates from the reference solution, indicating that the flow behavior is different. In the reference solution, the period after the bubble plume leaves the channel is where a straight vertical column of bubbly flow is observed. In the case of $\epsilon = 0.1$, the column of bubbly flow is not straight (Fig. 6.4) and the onset of precessing flow occurs much earlier than the other simulations.

The gas fraction is sampled along the line $y = 0.08\,\mathrm{m}$ and the profile along the x-axis is plotted in Fig. 6.11. This height corresponds to the widest part of the bubble plume. For

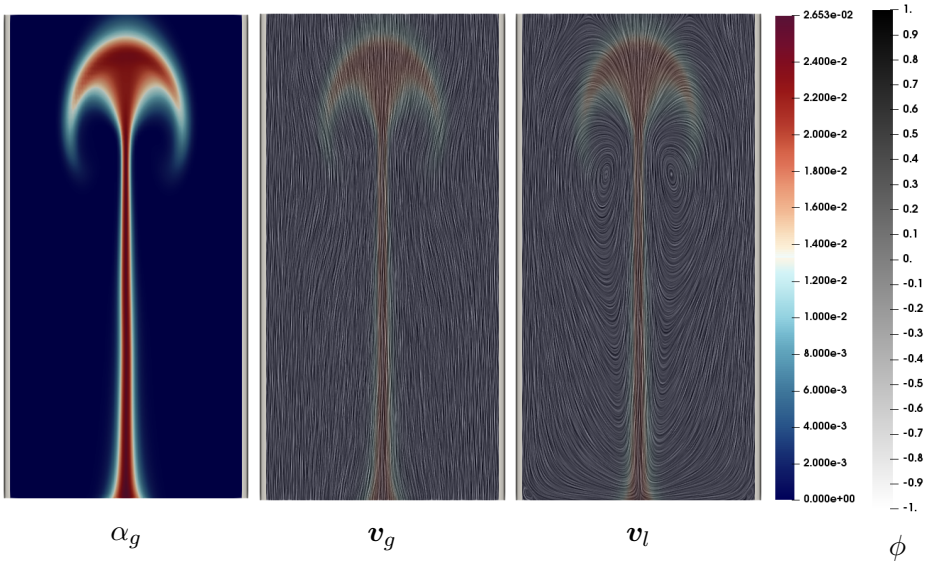


Figure 6.8: Surface plot of (left) α_g, (center) gas velocity and (right) liquid velocity at $t = 1.72\,\mathrm{s}$ with hyperbolic tangent diffuse-interface and $\epsilon = 0.01$.

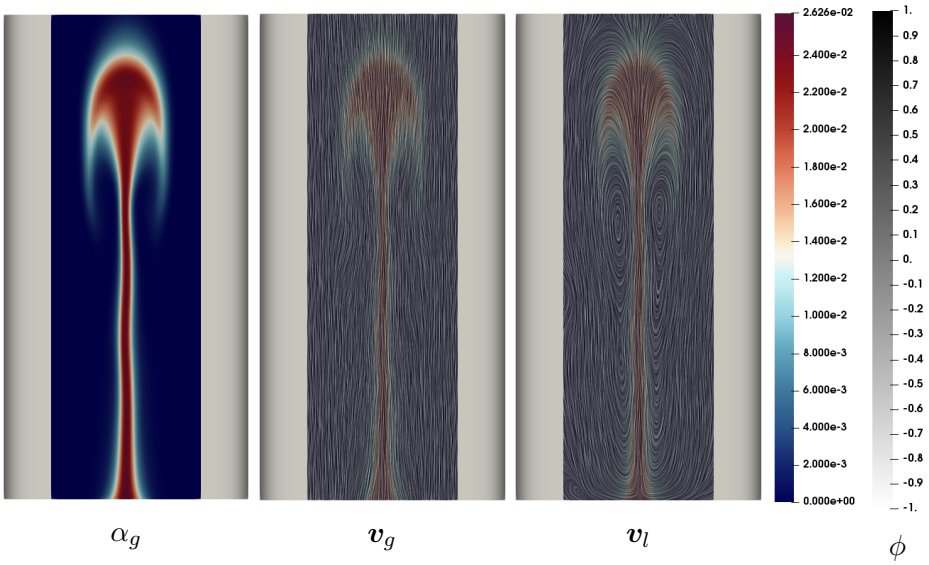


Figure 6.9: Surface plot of (left) α_g, (center) gas velocity and (right) liquid velocity at $t = 1.72\,\mathrm{s}$ with hyperbolic tangent diffuse-interface and $\epsilon = 0.1$.

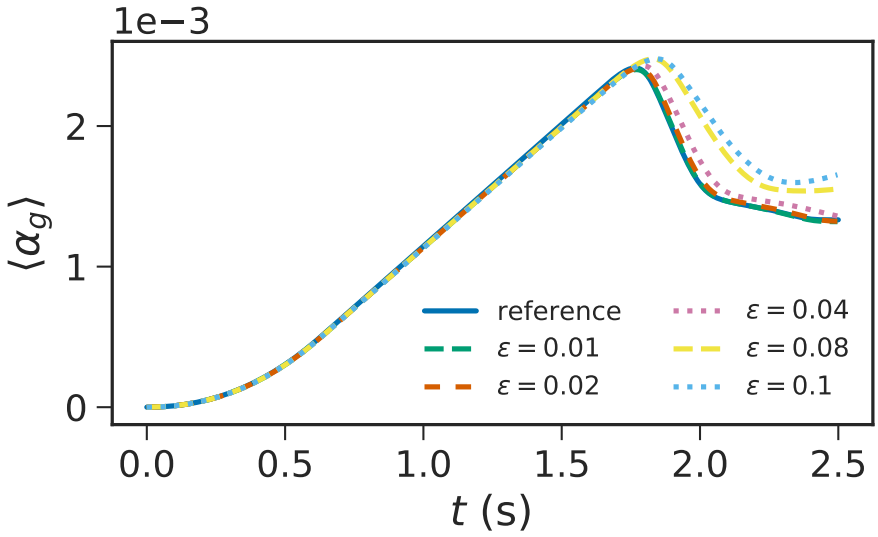
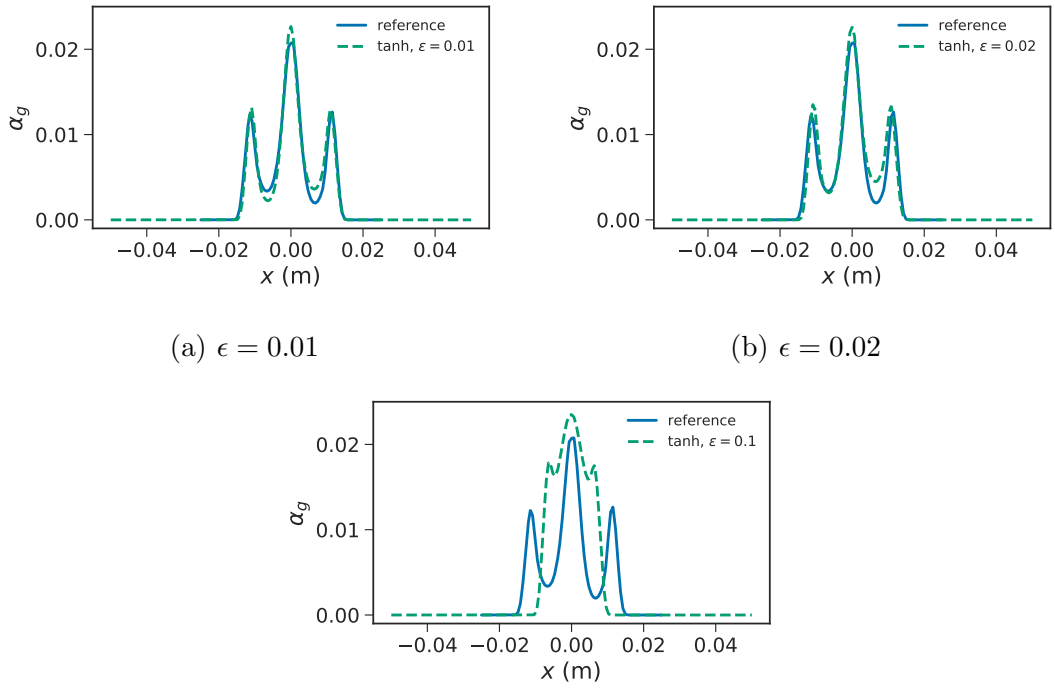


Figure 6.10: Time evolution of overall gas hold-up inside a channel with solid boundaries defined by a hyperbolic tangent diffuse-interface.

the cases where $\epsilon = 0.01$ and $\epsilon = 0.02$, the α_g profiles obtained using a diffuse-interface to impose solid boundaries show good qualitative agreement with the reference solution from Chapter 5. The agreement is improved as the interface becomes narrower but the difference is relatively small. As the interface becomes wider, the α_g profile is no longer in agreement with the reference solution. The effect of the diffuse-interface is also clear here as α_g in Fig. 6.11c starts to transition from $\alpha_g = 0$ to a nonzero value further into the domain.

To obtain a quantitative measure of how the diffuse-interface simulations compare with the reference solution, the width of the bubble plume at $y = 0.08\,\text{m}$ is computed and reported in Table 6.3. The plume widths from simulations with $\epsilon = 0.01$ and $\epsilon = 0.02$ are within 3% of the reference solution, supporting the observations made in this section. The plume width from $\epsilon = 0.1$ is 30% off from the reference solution, highlighting the importance of the diffuse-interface width.



(a) $\epsilon = 0.01$

(b) $\epsilon = 0.02$

(c) $\epsilon = 0.1$

Figure 6.11: α_g profile along $y = 0.08\,\text{m}$ with different hyperbolic tangent diffuse-interface widths.

Table 6.3: Bubble plume width at $y = 0.08\,\text{m}$ from simulations using hyperbolic tangent diffuse-interface.

Study	x_{plume} ($\times 10^{-2}$ m)	Difference (%)
Reference	3.21	–
$\epsilon = 0.01$	3.17	1.25
$\epsilon = 0.02$	3.12	2.80
$\epsilon = 0.04$	2.92	9.03
$\epsilon = 0.08$	2.51	21.8
$\epsilon = 0.1$	2.25	30.0

Effect of Interface Function

The previous studies have been conducted with a hyperbolic tangent function as the phase-field. Other functions can also be used to represent the diffuse-interface as long as it ensures a smooth transition from the solid to the fluid region. An example of this is a piece-wise cosine function where the interface region is described by a cosine function that is between $[-1, 1]$ and outside the interface region, $\phi = \pm 1$. Unlike the hyperbolic tangent function that asymptotically approaches the lower and upper bounds of ϕ, the piece-wise cosine function will reach $\phi = \pm 1$ exactly at the specified η, making it easier to control the interface width. In this section, the following piece-wise cosine function is used to impose the diffuse interface:

$$\phi(\tilde{\boldsymbol{x}}) = -\cos\left(-\pi \min\left[1, \max\left(0, \frac{|\tilde{x}| - \tilde{x}_c + 0.5\eta}{\eta}\right)\right]\right), \tag{6.28}$$

where ϕ will be ± 1 outside the region $\tilde{x} \in (\tilde{x}_c - 0.5\eta, \tilde{x}_c + 0.5\eta)$, depending on which side of the channel wall is $\tilde{x}$ close to.

Figure 6.12 shows the how ϕ varies with respect to x when defined using a hyperbolic tangent function, $\phi = \tanh(x/0.5\epsilon)$, and using a piece-wise cosine function centered at $x_c = 0$, $\phi = -\cos(-\pi \min[1, \max(0, (x + 0.5\eta)/\eta)])$, for a comparable interface width. The width of the cosine interface is approximated by $\eta = \epsilon \tanh^{-1}(0.999)$, which corresponds to the distance between $\phi = \pm 0.999$ in the hyperbolic tangent case. From Fig. 6.12, the transition of ϕ from -1 to 1 in the piece-wise cosine function is more gradual than the hyperbolic tangent function, which results in lower values of $\boldsymbol{\nabla}\phi$.

Figure 6.13 shows the gas fraction profile and velocity streamlines for simulations with a piece-wise cosine diffuse-interface with a comparable interface width as the hyperbolic tangent case. At small ϵ, the profiles are qualitatively similar to their hyperbolic tangent counterparts. The bubble plume in the $\epsilon = 0.1$ case is still noticeably narrower than the reference solution but appears to be wider than the hyperbolic tangent result with the same ϵ. The gas column below the plume also appears to be more stable than the results in Fig. 6.9.

The significant difference between the results from different interface functions at $\epsilon =$

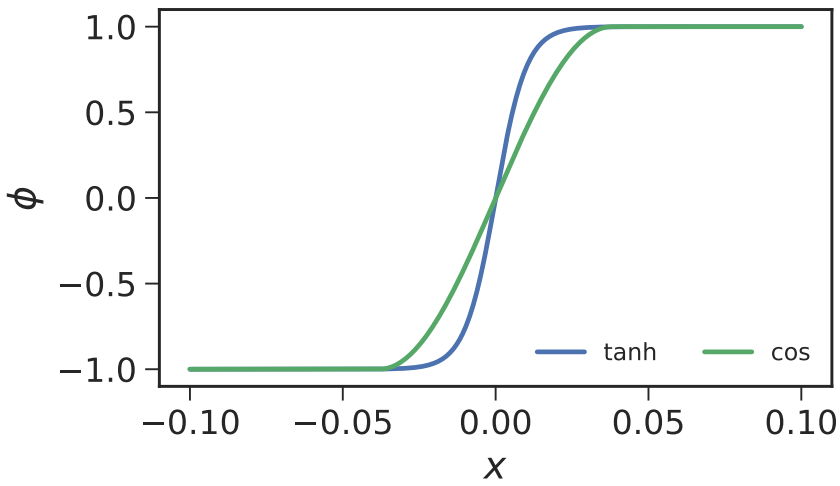


Figure 6.12: Comparison of diffuse-interface generated using hyperbolic tangent and piece-wise cosine functions with $\epsilon = 0.02$ and $\eta = \epsilon \tanh^{-1}(0.999)$.

0.1 is due to the piece-wise nature of Eqn. (6.28) and the approximation of the interface width to obtain a comparable width as the asymptotic hyperbolic tangent function. In the piece-wise cosine function, the approximation $\eta = \epsilon \tanh^{-1}(0.999)$ produces a diffuse-interface that approaches $\phi = \pm 1$ over a similar length-scale as the hyperbolic tangent function for small interface widths. However, at $\epsilon = 0.1$, the difference between $\epsilon \tanh^{-1}(0.999)$ and $\epsilon \tanh^{-1}(0.9999)$, which are interface widths approximated by $\phi = \pm 0.999$ and $\phi = \pm 0.9999$, respectively, is an order of magnitude larger than at $\epsilon = 0.01$ and non-negligible. The hyperbolic tangent function smears the interface over a larger distance, which for larger values of ϵ, is detrimental to the performance of the method.

Figure 6.14 shows the time evolution of the overall gas-holdup for simulations with a piece-wise cosine diffuse-interface. Similar to the hyperbolic tangent case, the gas hold-up at small interface widths ($\epsilon = 0.01$ and $\epsilon = 0.02$) are in agreement with the reference solution. At $\epsilon = 0.1$, the gas hold-up is not in agreement with the reference solution after the bubble plume starts to exit the simulation domain but the difference is not as drastic as the hyperbolic tangent case in Fig. 6.10.

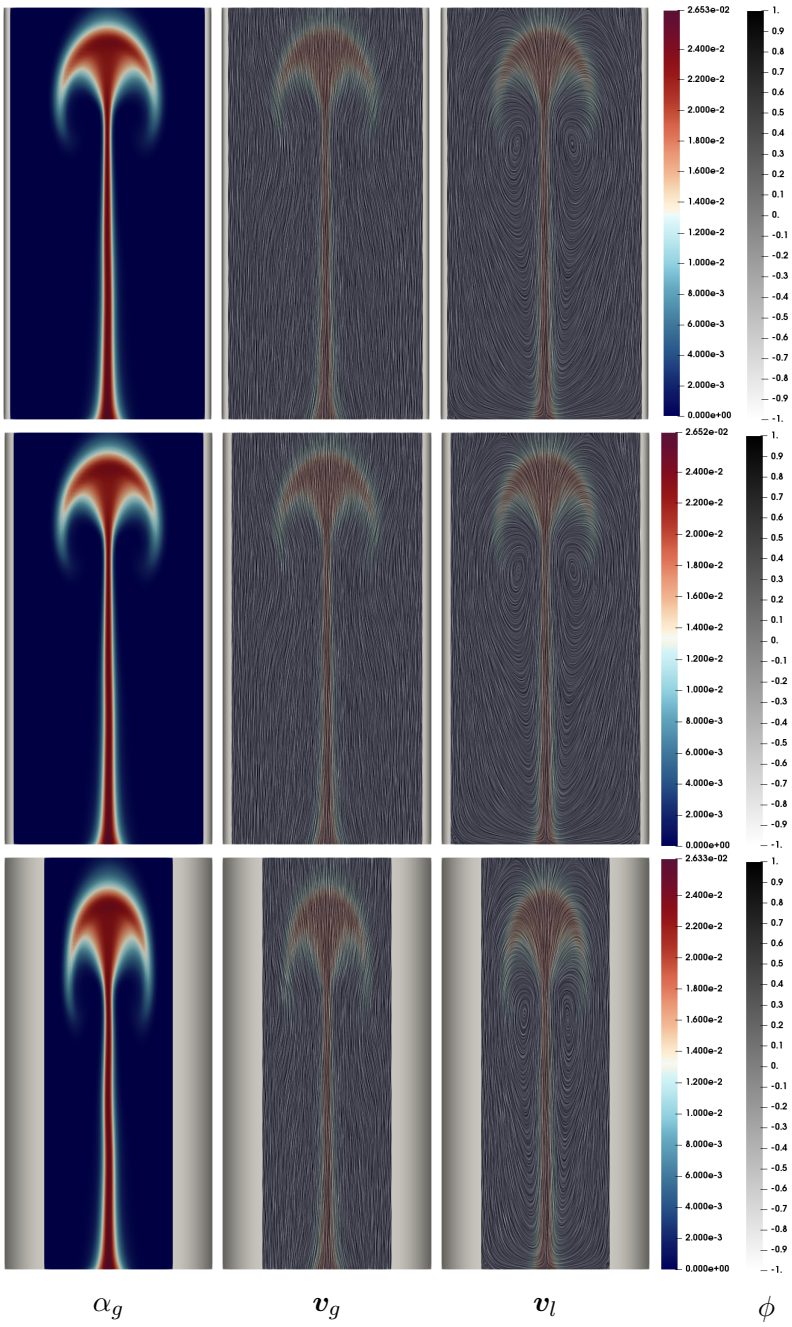


Figure 6.13: Surface plot of (left) phase fraction, (center) gas velocity and (right) liquid velocity at $t = 1.72\,\mathrm{s}$ with piece-wise cosine diffuse-interface and (top) $\epsilon = 0.01$, (middle) $\epsilon = 0.02$ and (bottom) $\epsilon = 0.1$.

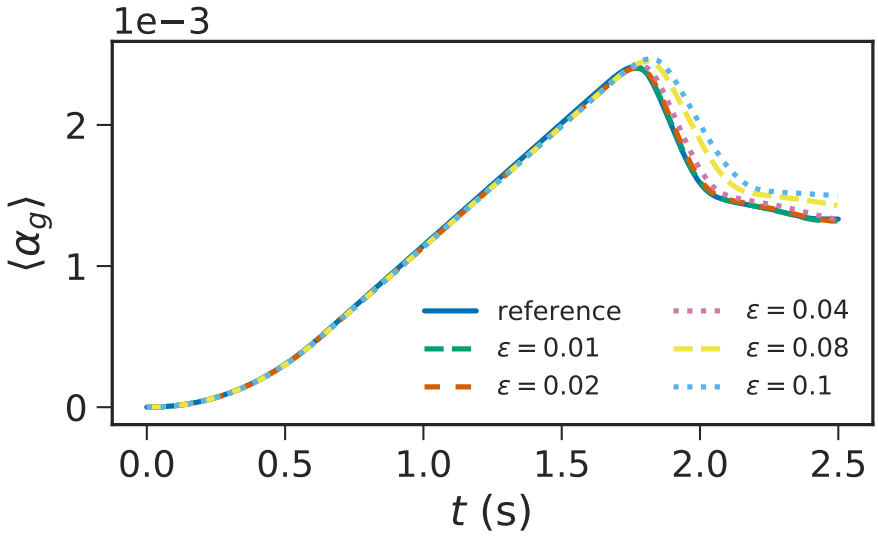


Figure 6.14: Time evolution of overall gas hold-up inside a channel with solid boundaries defined by a piece-wise cosine diffuse-interface.

The α_g profile at $y = 0.08\,\mathrm{m}$ from the three cases are plotted with the reference solution in Fig. 6.15. The results are similar to that observed in the previous section where $\epsilon = 0.01$ and $\epsilon = 0.02$ yielded profiles that are comparable to the reference solution but the profile from $\epsilon = 0.1$ is different from the reference solution. Figure 6.16 describes the error in the phase fraction along the line $y = 0.08\,\mathrm{m}$ as the interface width varies for both interface functions. The error is defined as:

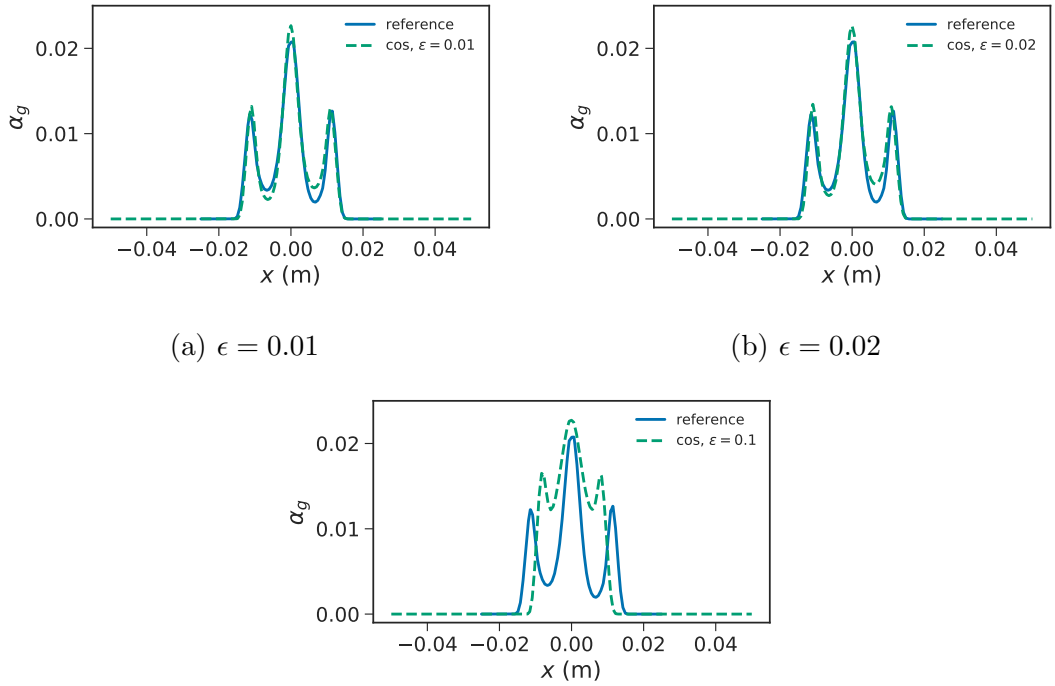
$$\text{Error} = \|\alpha_{g,ref} - \alpha_g\|_{y=0.08\,\mathrm{m}}, \tag{6.29}$$

and can be described using the following power-law expression:

$$\|\alpha_{g,ref} - \alpha_g\|_{y=0.08\,\mathrm{m}} = A\epsilon^m, \tag{6.30}$$

where A is a constant and m is the exponent. For both interface functions, the error follows an approximate first-order decay with the interface width where $m_{\tanh} = 0.953$ and $m_{\cos} = 0.896$.

The bubble plume width is computed and tabulated in Table 6.4. At $\epsilon = 0.01$, the bubble plume width is exactly the same as the hyperbolic tangent case. The $\epsilon = 0.02$ yielded a small difference between the two interface functions but is still below 3%. The use of the piece-wise cosine function as the interface improved the bubble plume width in the very diffuse case, decreasing the difference from the reference solution by almost 10%. This is due to the lack of smearing when ϕ is very close to ± 1 in the piece-wise cosine function compared to the hyperbolic tangent function.



(a) $\epsilon = 0.01$

(b) $\epsilon = 0.02$

(c) $\epsilon = 0.1$

Figure 6.15: α_g profile along $y = 0.08\,\text{m}$ with different piece-wise cosine diffuse-interface widths.

Table 6.4: Bubble plume width at $y = 0.08\,\mathrm{m}$ from simulations using piece-wise cosine diffuse-interface.

Study	x_{plume} ($\times 10^{-2}$ m)	Difference (%)
Reference	3.21	–
$\epsilon = 0.01$	3.17	1.25
$\epsilon = 0.02$	3.13	2.49
$\epsilon = 0.04$	3.03	5.46
$\epsilon = 0.08$	2.75	14.2
$\epsilon = 0.1$	2.54	20.9

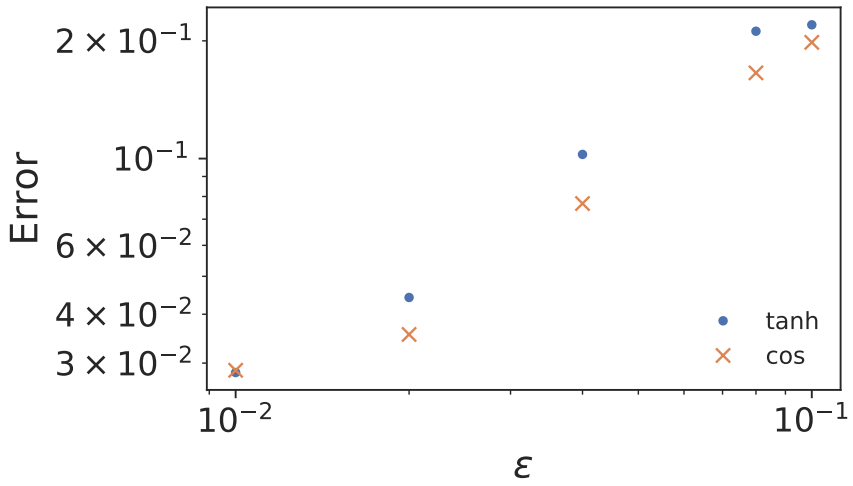


Figure 6.16: Error in α_g profile along $y = 0.08\,\mathrm{m}$ as a function of ϵ.

6.2.2 Flow Past a Cylinder

The diffuse-interface method is also used to model two-phase flow past a stationary cylinder for the first time with the two-fluid model. Simulations are performed using both the hyperbolic tangent and piece-wise cosine interface functions. For the hyperbolic tangent case, the cylinder is defined using the following function:

$$\phi(\tilde{\boldsymbol{x}}) = -\tanh\left(\frac{\|\tilde{\boldsymbol{x}} - \tilde{\boldsymbol{x}}_c\| - \tilde{R}}{0.5\epsilon}\right), \tag{6.31}$$

where $\tilde{\boldsymbol{x}}_c = (0, 0.8)$ is the scaled diffuse-interface position vector that corresponds to the center of the cylinder, $\tilde{R} = 0.1$ is the scaled radius of the cylinder and ϵ is chosen to be 0.005, 0.01 and 0.02. The piece-wise cosine interface is defined by:

$$\phi(\tilde{\boldsymbol{x}}) = -\cos\left(-\pi \min\left[1, \max\left(0, \frac{\|\tilde{\boldsymbol{x}} - \tilde{\boldsymbol{x}}_c\| - \tilde{R} + 0.5\eta}{\eta}\right)\right]\right), \tag{6.32}$$

where $\eta = \epsilon \tanh^{-1}(0.999)$. In this system, the presence of the diffuse-interface is expected to have a larger impact on the flow profile due to the fact that the cylinder is directly in the path of the flow. The diffuse-interface will smear the solid boundary, making the cylinder appear slightly larger than if the boundary was defined using the mesh, which in turn will affect the hydrodynamical behavior of the fluid.

Figure 6.17 shows the gas and liquid velocity streamlines along with the gas fraction profile from the reference solution. The reference solution is obtained from performing a simulation with the conditions outlined in Section 6.1.3 using a boundary-conformal mesh. In the early stages of the simulation, gas moves around the cylinder, leaving a small area behind the cylinder for liquid recirculation. As the gas travels further up the channel ($t = 3.13\,\mathrm{s}$), it merges behind the cylinder and moves up the channel. The there are two zones of liquid recirculation near the inlet, one on each side of the bubbly mixture. The recirculation zones grow in size and their center move upward ($t = 3.13\,\mathrm{s}$). Over time, more and more mixing occurs, resulting in a wavy column of bubbly mixture and a distorted bubble plume at the very top. Several recirculation zones are present on either side of the

wavy column where gas is occasionally pulled into the vortex, resulting in areas of higher gas fractions.

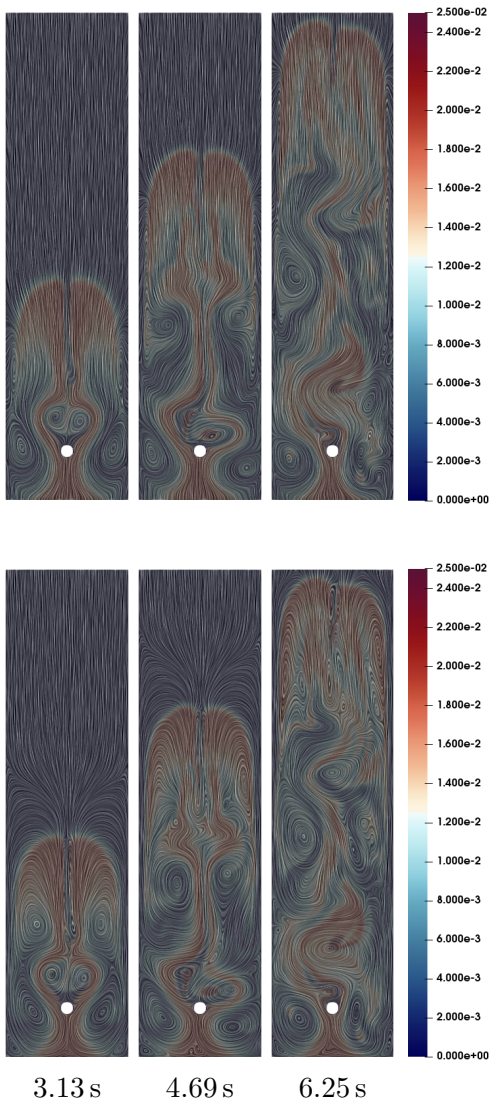


Figure 6.17: Evolution of gas-liquid flow past a stationary cylinder with a boundary-conformal mesh. Streamlines are of (top) gas and (bottom) liquid phases.

Figures 6.18 and 6.19 show the results at the same time steps from simulations using a hyperbolic tangent and piece-wise cosine diffuse-interfaces with $\epsilon = 0.01$, respectively.

The results do not appear to significantly differ when different interface functions are used. At $t = 3.13\,\mathrm{s}$, the gas phase fraction profile and the velocity streamlines appear to be the same as the results from the reference simulation for both interface functions. This is not the case starting at $t = 4.69\,\mathrm{s}$ onward. The recirculation zones in the wake of the cylinder predicted by the diffuse-interface simulations are wider and closer to the cylinder. This appears to have affected the evolution of the gas and velocity profiles, resulting in a similar features but different gas fraction and velocity profiles, confirming the prediction made earlier in this section.

The time evolution of the overall gas hold-up is shown in Fig. 6.20. In the early stages of the simulation, the hold-up evolves in the same manner as the reference solution. The interface function does not appear to significantly affect the solution at $\epsilon = 0.01$, supporting the results from Section 6.2.1. But as the diffuse-interface interacts with the flow, the gas hold-up diverges from the reference solution. This corresponds to the observations made in Figs. 6.17 to 6.19. While the magnitude and the slope of the gas hold-up profiles from the diffuse-interface simulations vary from the reference solution, the qualitative behavior is still the same.

6.3 Conclusions

A diffuse-interface method for imposing solid boundaries in two-phase flow has been developed. The Dirichlet solid boundary conditions are imposed by blending the governing equations of the two-fluid model with the Dirichlet boundary condition, resulting in a smooth transition from the solid boundary to the fluid domain. To validate the method, simulations of channel flow and flow past a cylinder are performed and the results are compared to results from simulations with boundary-conformal meshes. The results from diffuse-interface method for simulations of channel flow are found to be in agreement with the reference solution when the diffuse-interface is sufficiently small. At small interface widths, the choice of the interface function does not affect the accuracy of the solution. When the interface is large, the solution is negatively affected. In two-phase flow past a stationary cylinder, the results from the diffuse-interface simulations are in agreement

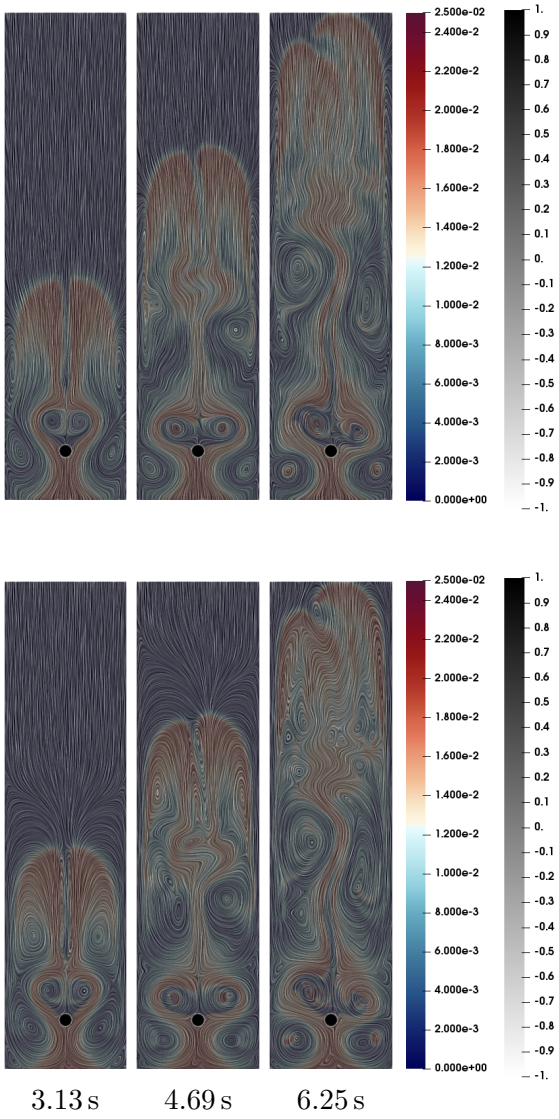


Figure 6.18: Evolution of gas-liquid flow past a stationary cylinder with a hyperbolic tangent diffuse-interface and $\epsilon = 0.01$. The diffuse-interface is in grayscale and streamlines are of (top) gas and (bottom) liquid velocities.

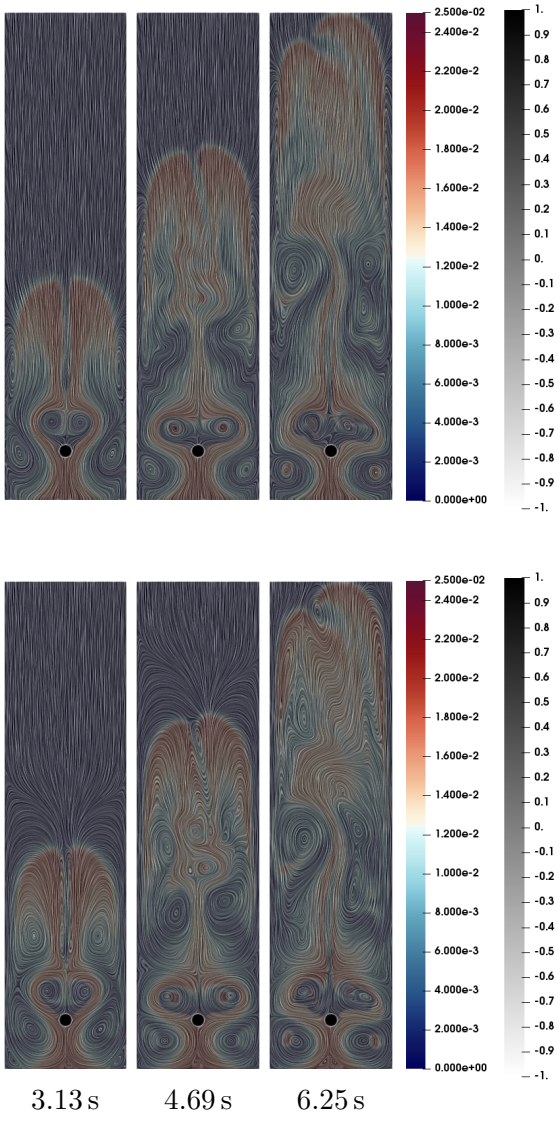


Figure 6.19: Evolution of gas-liquid flow past a stationary cylinder with a piece-wise cosine diffuse-interface and $\epsilon = 0.01$. The diffuse-interface is in grayscale and streamlines are of (top) gas and (bottom) liquid velocities.

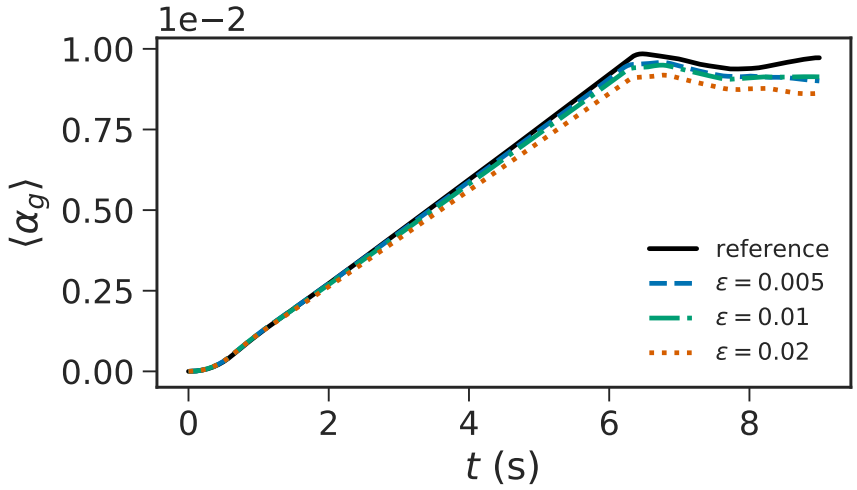


Figure 6.20: Time evolution of overall gas hold-up in flow past a stationary cylinder.

with the reference solution in the early stages of the simulation. As the flow interacts with the cylinder, the diffuse-interface is found to affect the flow profile and the overall gas hold-up. Similar flow features are still observed but at different locations and with different magnitudes.

Chapter 7

Conclusions and Recommendations for Future Work

7.1 Conclusions

A diffuse-interface method for imposing solid-fluid boundaries for two-phase bubbly flow using the two-fluid (Euler-Euler) model has been developed with implicitly-imposed phase fraction boundedness. Simulations using the diffuse-interface two-fluid method developed in this work are found to be in quantitative agreement with simulation results obtained using a conformal mesh to impose solid-fluid boundaries. The presented method allows for solid-fluid boundaries to be diffuse while still imposing the boundary conditions, which in turn allows complex geometries to be represented and easily modified during the course of the simulation.

The general conclusions of this work are:

- The use of an implicit nonlinear variational inequality solver can be used to impose phase fraction boundedness for the numerical solution of the two-fluid model without the introduction of *ad hoc* model contributions or post-processing.

- The effect of the assumption of the phasic pressure equality in the two-fluid model in dilute bubbly flow is non-negligible.
- Solid-fluid boundaries may be imposed in the two-fluid model through the use of a diffuse-interface method.
- The width and structure of the diffuse-interface has a significant effect on the accuracy of the imposed boundary conditions and consequently, the predicted flow profile as compared to the conformal mesh/grid numerical solution.
- The time evolution of the gas hold-up and the gas phase fraction profile from the diffuse-interface simulations of channel flow with sufficiently low interface widths are in agreement with the reference simulation performed without a diffuse-interface.

7.2 Recommendations for Future Work

The main set of recommendations for future work in this area are focused on *enhanced physical fidelity* of the model for dilute bubbly flows and include:

1. *Bubble size distribution* – At higher gas fractions, the bubbles will not be uniform in size, changing the behavior of the flow [152, 153]. This effect needs to be taken into account in the momentum exchange terms.
2. *Bubble coalescence and breakup* – As the bubbles move inside the domain, they could collide and coalesce or break into smaller bubbles [154–157]. This will alter the bubble size distribution and by extension, the interphase momentum exchange.
3. *Bubble swarming* – Large concentrations of bubbles within an area can have a swarming effect that alters the momentum exchange between the phases [13, 158, 159]. Taking this into account will improve the predictive nature of the simulations.
4. *Momentum exchange* – In this work, only the drag force is considered, however, there are other interphase momentum exchange contributions that can affect the simulation

results such as lift, virtual mass and wall lubrication force that should be considered if they apply to the system. The constitutive relationships for momentum exchange should also reflect the physics of the system and not be used as a tuning parameter to obtain agreement with experimental results and/or reference solution.

5. *Phase inversion* – The situation where two continuous phases exist in the simulation domain is also possible. This occurs in areas with higher gas fractions where the gas phase becomes the continuous phase in those areas. The interphase momentum exchange terms will be different when the continuous phase changes. This can be taken into account in many ways, including introducing blending in the momentum transfer terms [87] or using another diffuse-interface and solving two sets of governing equations.

In this work, the Dirichlet boundary conditions are enforced using a blending method. The drawback of this method is that it cannot impose Dirichlet boundary conditions on the normal or tangential component of a vector without *a priori* knowledge of the unit normal or tangent vector from the boundary. This can be addressed using Nitsche's method [56] to weakly impose the Dirichlet boundary condition.

Additionally, the diffuse-interface method developed in this work lays the groundwork for potential studies involving topological optimization of multiphase flow systems. In the future, an optimization routine can be incorporated to allow for the solid boundaries to be adjusted to optimize a specific objective function without modifying the mesh.

References

[1] H. A. Jakobsen, H. Lindborg and C. A. Dorao. Modeling of bubble column reactors: progress and limitations. *Ind. Eng. Chem. Res.* **44** (2005), 5107–5151.

[2] J. Joshi. Computational flow modelling and design of bubble column reactors. *Chem. Eng. Sci.* **56** (2001), 5893–5933.

[3] K. Ekambara, M. T. Dhotre and J. B. Joshi. CFD simulations of bubble column reactors: 1D, 2D and 3D approach. *Chem. Eng. Sci.* **60** (2005), 6733–6746.

[4] R. Krishna and J. van Baten. Scaling up bubble column reactors with the aid of CFD. *Chem. Eng. Res. Des.* **79** (2001), 283–309.

[5] S. Becker, A. Sokolichin and G. Eigenberger. Gas-liquid flow in bubble columns and loop reactors: Part II. comparison of detailed experiments and flow simulations. *Chem. Eng. Sci.* **49** (1994), 5747–5762.

[6] A. Sokolichin and G. Eigenberger. Gas-liquid flow in bubble columns and loop reactors: Part I. detailed modelling and numerical simulation. *Chem. Eng. Sci.* **49** (1994), 5735–5746.

[7] T. Frank, P. Zwart, E. Krepper, H.-M. Prasser and D. Lucas. Validation of CFD models for mono- and polydisperse air–water two-phase flows in pipes. *Nucl. Eng. Des.* **238** (2008), 647–659.

[8] T.-J. Chuang and T. Hibiki. Vertical upward two-phase flow CFD using interfacial area transport equation. *Prog. Nucl. Energy* **85** (2015), 415 –427.

[9] F. M. Erdal, S. A. Shirazi, O. Shoham and G. E. Kouba. CFD simulation of single-phase and two-phase flow in gas-liquid cylindrical cyclone separators. *SPE J.* **2** (1997), 436–446.

[10] R. Issa and M. Kempf. Simulation of slug flow in horizontal and nearly horizontal pipes with the two-fluid model. *Int. J. Multiphase Flow* **29** (2003), 69–95.

[11] R. I. Issa, M. Bonizzi and S. Barbeau. Improved closure models for gas entrainment and interfacial shear for slug flow modelling in horizontal pipes. *Int. J. Multiphase Flow* **32** (2006), 1287–1293.

[12] M. R. Ansari and V. Shokri. New algorithm for the numerical simulation of two-phase stratified gas–liquid flow and its application for analyzing the Kelvin-Helmholtz instability criterion with respect to wavelength effect. *Nucl. Eng. Des.* **237** (2007), 2302–2310.

[13] C. D. Lane, C. A. McKnight, J. Wiens, K. Reid and A. A. Donaldson. Parametric analysis of internal gas separation within an ebullated bed reactor. *Chem. Eng. Res. Des.* **105** (2016), 44–54.

[14] R. C. Chen, J. Reese and L.-S. Fan. Flow structure in a three-dimensional bubble column and three-phase fluidized bed. *AIChE J.* **40** (1994), 1093–1104.

[15] Y. T. Shah, B. G. Kelkar, S. P. Godbole and W.-D. Deckwer. Design parameters estimations for bubble column reactors. *AIChE J.* **28** (1982), 353–379.

[16] M. Ishii and T. Hibiki. *Thermo-Fluid Dynamics of Two-Phase Flow*. 2nd ed. Springer, 2011.

[17] H. A. Jakobsen. *Chemical Reactor Modeling: Multiphase Reactive Flows*. 2nd ed. Springer International Publishing, 2014.

[18] N. I. Kolev. *Multiphase Flow Dynamics – Fundamentals*. 4th ed. Vol. 1. Springer, 2011.

[19] D. L. Marchisio and R. O. Fox, eds. *Multiphase Reacting Flows: Modelling and Simulation*. Vol. 492. CISM International Centre for Mechanical Sciences. Springer Vienna, 2007.

[20] M. Agnaou, T. Treeratanaphitak, A. Mowla, M. Ioannidis, N. M. Abukhdeir and H. Budman. On the use of physical boundary conditions for two-phase flow simulations: integration of control feedback. *Comput. Chem. Eng.* (In Press).

[21] T. Borrvall and J. Petersson. Topology optimization of fluids in Stokes flow. *Int. J. Numer. Methods Fluids* **41** (2003), 77–107.

[22] Y. Deng, Z. Liu and Y. Wu. Topology optimization of steady and unsteady incompressible Navier-Stokes flows driven by body forces. *Struct. Multidiscip. Optim.* **47** (2013), 555–570.

[23] Y. Deng, Z. Liu, P. Zhang, Y. Liu and Y. Wu. Topology optimization of unsteady incompressible Navier-Stokes flows. *J. Comput. Phys.* **230** (2011), 6688 –6708.

[24] H. Garcke, C. Hecht, M. Hinze and C. Kahle. Numerical approximation of phase field based shape and topology optimization for fluids. *SIAM J. Sci. Comput.* **37** (2015), A1846–A1871.

[25] H. Garcke, M. Hinze, C. Kahle and K. F. Lam. A phase field approach to shape optimization in Navier–Stokes flow with integral state constraints. *Adv. Comput. Math.* (2018).

[26] S. Kreissl, G. Pingen and K. Maute. Topology optimization for unsteady flow. *Int. J. Numer. Meth. Eng* **87** (2011), 1229–1253.

[27] K. Maute. "Topology optimization of flows: stokes and navier-stokes models." *Topology Optimization in Structural and Continuum Mechanics*. Ed. by G. I. N. Rozvany and T. Lewiński. Vienna: Springer Vienna, 2014, 409–419.

[28] A. Sokolichin, G. Eigenberger and A. Lapin. Simulation of buoyancy driven bubbly flow: Established simplifications and open questions. *AIChE J.* **50** (2004), 24–45.

[29] M. V. Tabib, S. A. Roy and J. B. Joshi. CFD simulation of bubble column – An analysis of interphase forces and turbulence models. *Chem. Eng. J.* **139** (2008), 589–614.

[30] A. Lapin and A. Lübbert. Numerical simulation of the dynamics of two-phase gas-liquid flows in bubble columns. *Chem. Eng. Sci.* **49** (1994), 3661–3674.

[31] A. Sokolichin, G. Eigenberger, A. Lapin and A. Lübert. Dynamic numerical simulation of gas-liquid two-phase flows Euler/Euler versus Euler/Lagrange. *Chem. Eng. Sci.* **52** (1997), 611–626.

[32] R. Groll, S. Jakirlić and C. Tropea. Comparative study of Euler/Euler and Euler/Lagrange approaches simulating evaporation in a turbulent gas-liquid flow. *Int. J. Numer. Methods Fluids* **59** (2009), 873–906.

[33] S. Subramaniam. Lagrangian-Eulerian methods for multiphase flows. *Prog. Energy Combust. Sci.* **39** (2013), 215–245.

[34] C. Hirt and B. Nichols. Volume of fluid (VOF) method for the dynamics of free boundaries. *J. Comput. Phys.* **39** (1981), 201–225.

[35] G. Tryggvason, R. Scardovelli and S. Zaleski. *Direct Numerical Simulations of Gas-Liquid Multiphase Flows*. Cambridge University Press, 2011.

[36] H. Abels, H. Garcke and G. Grün. Thermodynamically consistent, frame indifferent diffuse interface models for incompressible two-phase flows with different densities. *Math. Mod. Meth. Appl. S.* **22** (2012), 1150013.

[37] H. Ding, P. D. M. Spelt and C. Shu. Diffuse interface model for incompressible two-phase flows with large density ratios. *J. Comput. Phys.* **226** (2007), 2078–2095.

[38] D. Jacqmin. Calculation of two-phase Navier-Stokes flows using phase-field modeling. *J. Comput. Phys.* **155** (1999), 96–127.

[39] C. Liu, J. Shen and X. Yang. Decoupled energy stable schemes for a phase-field model of two-phase incompressible flows with variable density. *J. Sci. Comput.* **62** (2015), 601–622.

[40] J. Shen and X. Yang. A phase-field model and its numerical approximation for two-phase incompressible flows with different densities and viscosities. *SIAM J. Sci. Comput.* **32** (2010), 1159–1179.

[41] Y. Sun and C. Beckermann. Sharp interface tracking using the phase-field equation. *J. Comput. Phys.* **220** (2007), 626–653.

[42] N. Takada, M. Misawa and A. Tomiyama. A phase-field method for interface-tracking simulation of two-phase flows. *Math. Comput. Simulat.* **72** (2006), 220–226.

[43] D. M. Anderson, G. B. McFadden and A. A. Wheeler. Diffuse-interface methods in fluid mechanics. *Annu. Rev. Fluid Mech.* **30** (1998), 139–165.

[44] R. Gupta, D. F. Fletcher and B. S. Haynes. On the CFD modelling of Taylor flow in microchannels. *Chem. Eng. Sci.* **64** (2009), 2941–2950.

[45] R. M. Santos and M. Kawaji. Numerical modeling and experimental investigation of gas-liquid slug formation in a microchannel T-junction. *Int. J. Multiphase Flow* **36** (2010), 314–323.

[46] B. Bunner and G. Tryggvason. Direct numerical simulations of three-dimensional bubbly flows. *Phys. Fluids* **11** (1999), 1967–1969.

[47] M. van Sint Annaland, N. G. Deen and J. A. M. Kuipers. Numerical simulation of gas bubbles behaviour using a three-dimensional volume of fluid method. *Chem. Eng. Sci.* **60** (2005), 2999–3011.

[48] J. Donea, A. Huerta, J. Ponthot and A. Rodríguez-Ferran. "Arbitrary Lagrangian–Eulerian methods." *Encyclopedia of Computational Mechanics*. American Cancer Society, 2004. Chap. 14.

[49] R. Glowinski, T.-W. Pan, T. Hesla and D. Joseph. A distributed Lagrange multiplier/fictitious domain method for particulate flows. *Int. J. Multiphase Flow* **25** (1999), 755 –794.

[50] R. Mittal and G. Iaccarino. Immersed boundary methods. *Annu. Rev. Fluid Mech.* **37** (2005), 239–261.

[51] F. Sotiropoulos and X. Yang. Immersed boundary methods for simulating fluid–structure interaction. *Prog. Aerosp. Sci.* **65** (2014), 1 –21.

[52] I. Ramière, P. Angot and M. Belliard. A fictitious domain approach with spread interface for elliptic problems with general boundary conditions. *Comput. Methods Appl. Mech. Engrg.* **196** (2007), 766–781.

[53] X. Li, J. Lowengrub, A. Rätz and A. Voigt. Solving PDEs in complex geometries: a diffuse domain approach. *Commun. Math. Sci.* **7** (2009), 81–107.

[54] S. Aland, J. Lowengrub and A. Voigt. Two-phase flow in complex geometries: a diffuse domain approach. *Comput. Model Eng. Sci.* **57** (2010), 77–106.

[55] M. Schlottbom. Error analysis of a diffuse interface method for elliptic problems with Dirichlet boundary conditions. *Appl. Numer. Math.* **109** (2016), 109–122.

[56] L. H. Nguyen, S. K. Stoter, M. Ruess, M. A. Sanchez Uribe and D. Schillinger. The diffuse Nitsche method: Dirichlet constraints on phase-field boundaries. *Int. J. Numer. Meth. Eng* **113** (2018), 601–633.

[57] J. K. Patel and G. Natarajan. Diffuse interface immersed boundary method for multi-fluid flows with arbitrarily moving rigid bodies. *J. Comput. Phys.* **360** (2018), 202 –228.

[58] T. J. Hanratty. *Physics of Gas-Liquid Flows*. Cambridge University Press, 2013.

[59] Y. Taitel and A. E. Dukler. A model for predicting flow regime transitions in horizontal and near horizontal gas-liquid flow. *AIChE J.* **22** (1976), 47–55.

[60] J. M. Mandhane, G. A. Gregory and K. Aziz. A flow pattern map for gas-liquid flow in horizontal pipes. *Int. J. Multiphase Flow* **1** (1974), 537–553.

[61] K. Mishima and M. Ishii. Flow regime transition criteria for upward two-phase flow in vertical tubes. *Int. J. Heat Mass Transfer* **27** (1984), 723–737.

[62] J.-P. Zhang, J. Grace, N. Epstein and K. Lim. Flow regime identification in gas-liquid flow and three-phase fluidized beds. *Chem. Eng. Sci.* **52** (1997), 3979–3992.

[63] R. T. Lahey Jr. The simulation of multidimensional multiphase flows. *Nucl. Eng. Des.* **235** (2005), 1043–1060.

[64] D. A. Drew and S. L. Passman. *Theory of Multicomponent Fluids*. Vol. 135. Applied Mathematical Sciences. New York: Springer, 1998.

[65] M. Ishii and K. Mishima. Two-fluid model and hydrodynamic constitutive relations. *Nucl. Eng. Des.* **82** (1984), 107–126.

[66] R. T. Lahey Jr. and D. A. Drew. The analysis of two-phase flow and heat transfer using a multidimensional, four field, two-fluid model. *Nucl. Eng. Des.* **204** (2001), 29–44.

[67] S. P. Antal, R. T. Lahey Jr. and J. E. Flaherty. Analysis of phase distribution in fully developed laminar bubbly two-phase flow. *Int. J. Multiphase Flow* **17** (1991), 635–652.

[68] A. Tomiyama, H. Tamai, I. Zun and S. Hosokawa. Transverse migration of single bubbles in simple shear flows. *Chem. Eng. Sci.* **57** (2002), 1849–1858.

[69] D. A. Drew and R. T. Lahey Jr. The virtual mass and lift force on a sphere in rotating and straining inviscid flow. *Int. J. Multiphase Flow* **13** (1987), 113–121.

[70] D. A. Drew and R. T. Lahey Jr. "Analytical modeling of multiphase flow." *Particulate Two-Phase Flow*. Ed. by M. C. Roco. Butterworth-Heinemann Series in Chemical Engineering. Butterworth-Heinemann, 1993. Chap. 16, 509–566.

[71] J. H. Stuhmiller. The influence of interfacial pressure forces on the character of two-phase flow model equations. *Int. J. Multiphase Flow* **3** (1977), 551 –560.

[72] C. Pauchon and S. Banerjee. Interphase momentum interaction effects in the averaged multifield model: Part I: void propagation in bubbly flows. *Int. J. Multiphase Flow* **12** (1986), 559–573.

[73] R. G. Rice and D. D. Do. *Applied mathematics and modeling for chemical engineers*. 2nd. Wiley, 2012.

[74] B. A. Finlayson. *The Method of Weighted Residuals and Variational Principles*. Vol. 87. Mathematics in Science and Engineering. New York: Academic Press, 1972.

[75] E. B. Becker, G. F. Carey and J. T. Oden. *Finite Elements: An Introduction*. Vol. 1. Texas Finite Element Series. Prentice-Hall, 1981.

[76] J. Donea and A. Huerta. *Finite Element Methods for Flow Problems*. John Wiley & Sons, Inc., 2003.

[77] W. E. Schiesser and G. W. Griffiths. *A Compendium of Partial Differential Equation Models: Method of Lines Analysis with Matlab*. Cambridge University Press, 2009.

[78] H. K. Versteeg and W. Malalasekera. *An introduction to computational fluid dynamics – The finite volume method*. Longman Scientific & Technical, 1995.

[79] R. B. Bird, W. E. Stewart and E. N. Lightfoot. *Transport Phenomena*. 2nd revised. John Wiley & Sons, Inc., 2007.

[80] A. J. Chorin. A numerical method for solving incompressible viscous flow problems. *J. Comput. Phys.* **2** (1967), 12–26.

[81] A. J. Chorin. Numerical solution of the Navier-Stokes equations. *Math. Comp.* **22** (1968), 745–762.

[82] A. J. Chorin. On the convergence of discrete approximations to the navier-stokes equations. *Math. Comp.* **23** (1969), 341–353.

[83] S. Patankar and D. Spalding. A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. *Int. J. Heat Mass Transfer* **15** (1972), 1787–1806.

[84] R. Issa. Solution of the implicitly discretised fluid flow equations by operator-splitting. *J. Comput. Phys.* **62** (1985), 40–65.

[85] J. H. Ferziger and M. Perić. *Computational methods for fluid dynamics*. 3rd. Springer, 2002.

[86] C. Caia and P. Minev. A finite element method for an averaged multiphase flow model. *Int. J. Comput. Fluid D.* **18** (2004), 111–123.

[87] H. Weller. *Derivation, Modelling and Solution of the Conditionally Averaged Two-Phase Flow Equations*. Tech. rep. Technical Report TR/HGW/02, Nabla Ltd, 2005.

[88] A. D. Gosman, C. Lekakou, S. Politis, R. I. Issa and M. K. Looney. Multidimensional modeling of turbulent two-phase flows in stirred vessels. *AIChE J.* **38** (1992), 1946–1956.

[89] ANSYS, Inc. *ANSYS Fluent Theory Guide*. 2016.

[90] T. Uchiyama. Petrov-Galerkin finite element method for gas-liquid two-phase flow based on an incompressible two-fluid model. *Nucl. Eng. Des.* **193** (1999), 145–157.

[91] M. Giordano and V. Magi. Petrov-Galerkin finite element stabilization for two-phase flows. *Int. J. Numer. Methods Fluids* **51** (2006), 1117–1129.

[92] ANSYS, Inc. *ANSYS CFX Solver Theory Guide*. 2013.

[93] COMSOL. *CFD Module User's Guide*. 2015.

[94] Y. Bartosiewicz, J. Laviéville and J.-M. Seynhaeve. A first assessment of the NEPTUNE_CFD code: instabilities in a stratified flow comparison between the VOF method and a two-field approach. *Int. J. Heat Fluid Flow* **29** (2008), 460–478.

[95] S. Mimouni, M. Boucker, J. Laviéville, A. Guelfi and D. Bestion. Modelling and computation of cavitation and boiling bubbly flows with the NEPTUNE_CFD code. *Nucl. Eng. Des.* **238** (2008), 680–692.

[96] H. G. Weller, G. Tabor, H. Jasak and C. Fureby. A tensorial approach to computational continuum mechanics using object-oriented techniques. *Comput. Phys.* **12** (1998), 620–631.

[97] R. Rzehak and S. Kriebitzsch. Multiphase CFD-simulation of bubbly pipe flow: a code comparison. *Int. J. Multiphase Flow* **68** (2015), 135 –152.

[98] S. M. Damián. "An Extended Mixture Model for the Simultaneous Treatment of Short and Long Scale Interfaces." PhD thesis. Universidad Nacional del Litoral, 2013.

[99] ANSYS, Inc. *ANSYS Fluent User Guide*. 2016.

[100] The OpenFOAM Foundation. *OpenFOAM: Open Source Field Operation and Manipulation Library*. 2017.

[101] P. J. Oliveira and R. I. Issa. Numerical aspects of an algorithm for the eulerian simulation of two-phase flows. *Int. J. Numer. Methods Fluids* **43** (2003), 1177–1198.

[102] H. Rusche. "Computational Fluid Dynamics of Dispersed Two-Phase Flows at High Phase Fractions." PhD thesis. Imperial College, 2002.

[103] T. N. Dinh, R. R. Nourgaliev and T. G. Theofanous. "Understanding the ill-posed two-fluid model." *10th International Topical Meeting on Nuclear Reactor Thermal Hydraulics (NURETH-10), Seoul, Korea*. 2003, 5–9.

[104] A. Vaidheeswaran. "Well-Posedness and Convergence of CFD Two-Fluid Model for Bubbly Flows." PhD thesis. Purdue University, 2015.

[105] M. López de Bertodano, W. Fullmer, A. Clausse and V. H. Ransom. *Two-Fluid Model Stability, Simulation and Chaos*. Springer International Publishing, 2016.

[106] T. C. Haley, D. A. Drew and R. T. Lahey Jr. An analysis of the eigenvalues of bubbly two-phase flows. *Chem. Eng. Commun.* **106** (1991), 93–117.

[107] R. T. Lahey Jr., L. Y. Cheng, D. A. Drew and J. E. Flaherty. The effect of virtual mass on the numerical stability of accelerating two-phase flows. *International Journal of Multiphase Flow* **6** (1980), 281 –294.

[108] M.-S. Liou, L. Nguyen, C.-H. Chang, S. Sushchikh, R. Nourgaliev and T. Theofanous. "Hyperbolicity, discontinuities, and numerics of two-fluid models." *Computational Fluid Dynamics 2006*. Ed. by H. Deconinck and E. Dick. Springer Berlin Heidelberg, 2006, 625–630.

[109] S.-J. Lee, K.-S. Chang and S.-J. Kim. Surface tension effect in the two-fluids equation system. *Int. J. Heat Mass Transfer* **41** (1998), 2821–2826.

[110] A. Vaidheeswaran and M. L. de Bertodano. Interfacial pressure coefficient for ellipsoids and its effect on the two-fluid model eigenvalues. *J. Fluids Eng.* **138** (2016), 081302.

[111] J. H. Song and M. Ishii. The well-posedness of incompressible one-dimensional two-fluid model. *Int. J. Heat Mass Transfer* **43** (2000), 2221–2231.

[112] A. Vaidheeswaran, W. D. Fullmer and M. L. de Bertodano. Effect of collision force on well-posedness and stability of the two-fluid model for vertical bubbly flows. *Nucl. Sci. Eng.* **184** (2016), 353–362.

[113] I. Toumi and A. Kumbaro. An approximate linearized Riemann solver for a two-fluid model. *J. Comput. Phys.* **124** (1996), 286–300.

[114] J.-W. Park, D. A. Drew and R. Lahey Jr. The analysis of void wave propagation in adiabatic monodispersed bubbly two-phase flows using an ensemble-averaged two-fluid model. *Int. J. Multiphase Flow* **24** (1998), 1205–1244.

[115] C. S. Peskin. Flow patterns around heart valves: a numerical method. *J. Comput. Phys.* **10** (1972), 252–271.

[116] K. Khadra, P. Angot, S. Parneix and J. Caltagirone. Fictitious domain approach for numerical modelling of Navier–Stokes equations. *Int. J. Numer. Methods Fluids* **34** (2000), 651–684.

[117] L. Shen and E.-S. Chan. Numerical simulation of fluid-structure interaction using a combined volume of fluid and immersed boundary method. *Ocean Eng.* **35** (2008), 939–952.

[118] L. Shen and E.-S. Chan. Application of a combined IB-VOF model to wave–structure interactions. *Appl. Ocean Res.* **32** (2010), 40–48.

[119] L. Shen and E.-S. Chan. Numerical simulation of nonlinear dispersive waves propagating over a submerged bar by IB-VOF model. *Ocean Eng.* **38** (2011), 319–328.

[120] C. Zhang, W. Zhang, N. Lin, Y. Tang, C. Zhao, J. Gu, W. Lin, X. Chen and A. Qiu. A two-phase flow model coupling with volume of fluid and immersed boundary methods for free surface and moving structure problems. *Ocean Eng.* **74** (2013), 107–127.

[121] C. Zhang, N. Lin, Y. Tang and C. Zhao. A sharp interface immersed boundary/VOF model coupled with wave generating and absorbing options for wave-structure interaction. *Comput. Fluids* **89** (2014), 214–231.

[122] S. Gsell, T. Bonometti and D. Astruc. A coupled volume-of-fluid/immersed-boundary method for the study of propagating waves over complex-shaped bottom: application to the solitary wave. *Comput. Fluids* **131** (2016), 56–65.

[123] J. Yang and F. Stern. Sharp interface immersed-boundary/level-set method for wave-body interactions. *J. Comput. Phys.* **228** (2009), 6590–6616.

[124] Y. Suh and G. Son. A sharp-interface level-set method for simulation of a piezoelectric inkjet process. *Numer. Heat Tr. B-Fund.* **55** (2009), 295–312.

[125] M. Arienti and M. Sussman. An embedded level set method for sharp-interface multiphase simulations of diesel injectors. *Int. J. Multiphase Flow* **59** (2014), 1–14.

[126] H. V. Patel, S. Das, J. A. M. Kuipers, J. T. Padding and E. A.J. F. Peters. A coupled volume of fluid and immersed boundary method for simulating 3D multiphase flows with contact line dynamics in complex geometries. *Chem. Eng. Sci.* **166** (2017), 28–41.

[127] S. Vincent, A. Sarthou, J.-P. Caltagirone, F. Sonilhac, P. Février, C. Mignot and G. Pianet. Augmented Lagrangian and penalty methods for the simulation of two-phase flows interacting with moving solids. application to hydroplaning flows interacting with real tire tread patterns. *J. Comput. Phys.* **230** (2011), 956–983.

[128] S. Osher and R. P. Fedkiw. Level set methods: an overview and some recent results. *J. Comput. Phys.* **169** (2001), 463–502.

[129] T. Yabe, F. Xiao and T. Utsumi. The constrained interpolation profile method for multiphase analysis. *J. Comput. Phys.* **169** (2001), 556–593.

[130] G. Son. A level set method for incompressible two-fluid flows with immersed solid boundaries. *Numer. Heat Tr. B-Fund.* **47** (2005), 473–489.

[131] K. Washino, H. S. Tan, A. D. Salman and M. J. Hounslow. Direct numerical simulation of solid-liquid-gas three-phase flow: fluid-solid interaction. *Powder Technol.* **206** (2011), 161–169.

[132] X. Sun and M. Sakai. Three-dimensional simulation of gas-solid-liquid flows using the DEM-VOF method. *Chem. Eng. Sci.* **134** (2015), 531–548.

[133] P. Horgue, M. Prat and M. Quintard. A penalization technique applied to the "volume-of-fluid" method: wettability condition on immersed boundaries. *Comput. Fluids* **100** (2014), 255–266.

[134] X. Sun and M. Sakai. Numerical simulation of two-phase flows in complex geometries by using the volume-of-fluid/immersed-boundary method. *Chem. Eng. Sci.* **139** (2016), 221–240.

[135] N. G. Deen, M. van Sint Annaland and J. A. M. Kuipers. Direct numerical simulation of complex multi-fluid flows using a combined front trackingand immersed boundary method. *Chem. Eng. Sci.* **64** (2009), 2186–2201.

[136] H. S. Yoon, C. H. Jeon, J. H. Jung, B. Koo, C. Choi and S. C. Shin. Simulation of two-phase flow–body interaction problems using direct forcing/fictitious domain–level set method. *Int. J. Numer. Methods Fluids* **73** (2013), 250–265.

[137] A. Calderer, S. Kang and F. Sotiropoulos. Level set immersed boundary method for coupled simulation of air/water interaction with complex floating structures. *J. Comput. Phys.* **277** (2014), 201–227.

[138] K. Goda. A multistep technique with implicit difference schemes for calculating two- or three-dimensional cavity flows. *J. Comput. Phys.* **30** (1979), 76–95.

[139] A. Logg, K.-A. Mardal and G. Wells, eds. *Automated Solution of Differential Equations by the Finite Element Method*. Vol. 84. Lecture Notes in Computational Science and Engineering. Springer Berlin Heidelberg, 2012.

[140] U. Ascher and L. Petzold. *Computer methods for ordinary differential equations and differential-algebraic equations*. Society for Industrial Mathematics, 1998.

[141] A. Behzadi, R. Issa and H. Rusche. Modelling of dispersed bubble and droplet flow at high phase fractions. *Chem. Eng. Sci.* **59** (2004), 759–770.

[142] C. Geuzaine and J.-F. Remacle. Gmsh: a 3-D finite element mesh generator with built-in pre- and post-processing facilities. *Int. J. Numer. Meth. Eng* **79** (2009), 1309–1331.

[143] L. Schiller and A. Naumann. A drag coefficient correlation. *Z. Ver. Dtsch. Ing.* **77** (1935), 138.

[144] A. N. Brooks and T. J. R. Hughes. Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. *Comput. Methods Appl. Mech. Engrg.* **32** (1982), 199–259.

[145] S. Balay, S. Abhyankar, M. F. Adams, J. Brown, P. Brune, K. Buschelman, L. Dalcin, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, D. A. May, L. C. McInnes, K. Rupp, P. Sanan, B. F. Smith, S. Zampini, H. Zhang and H. Zhang. *PETSc Users Manual*. Tech. rep. Argonne National Laboratory, 2017.

[146] T. S. Munson, F. Facchinei, M. C. Ferris, A. Fischer and C. Kanzow. The semismooth algorithm for large scale complementarity problems. *INFORMS Journal on Computing* **13** (2001), 294–311.

[147] S. J. Benson and T. S. Munson. Flexible complementarity solvers for large-scale applications. *Optim. Method. Softw.* **21** (2006), 155–168.

[148] T. S. Munson, J. Sarich, S. Wild, S. J. Benson and L. C. McInnes. *TAO Users Manual*. Tech. rep. Argonne National Laboratory, 2017.

[149] M. C. Ruzicka and N. H. Thomas. Buoyancy-driven instability of bubbly layers: analogy with thermal convection. *Int. J. Multiphase Flow* **29** (2003), 249–270.

[150] R. F. Mudde. Gravity-driven bubbly flows. *Annu. Rev. Fluid Mech.* **37** (2005), 393–423.

[151] R. Peyret. *Spectral Methods for Incompressible Viscous Flow*. Vol. 148. Applied Mathematical Sciences. Springer-Verlag New York, 2002.

[152] A. K. Nayak, Z. Borka, L. E. Patruno, F. Sporleder, C. A. Dorao and H. A. Jakobsen. A combined multifluid-population balance model for vertical gas-liquid bubble-driven flows considering bubble column operating conditions. *Ind. Eng. Chem. Res.* **50** (2011), 1786–1798.

[153] P. Chen, J. Sanyal and M. Dudukovic. CFD modeling of bubble columns flows: implementation of population balance. *Chem. Eng. Sci.* **59** (2004), 5201–5207.

[154] M. R. Bhole, J. B. Joshi and D. Ramkrishna. CFD simulation of bubble columns incorporating population balance modeling. *Chem. Eng. Sci.* **63** (2008), 2267–2282.

[155] E. Krepper, D. Lucas, T. Frank, H.-M. Prasser and P. J. Zwart. The inhomogeneous MUSIG model for the simulation of polydispersed flows. *Nucl. Eng. Des.* **238** (2008), 1690–1702.

[156] E. Krepper, D. Lucas and H.-M. Prasser. On the modelling of bubbly flow in vertical pipes. *Nucl. Eng. Des.* **235** (2005), 597–611.

[157] M. Sattar, J. Naser and G. Brooks. Numerical simulation of two-phase flow with bubble break-up and coalescence coupled with population balance modeling. *Chem. Eng. Process. Process Intensif.* **70** (2013), 66–76.

[158] M. Simonnet, C. Gentric, E. Olmos and N. Midoux. CFD simulation of the flow field in a bubble column reactor: importance of the drag force formulation to describe regime transitions. *Chem. Eng. Process. Process Intensif.* **47** (2008), 1726–1737.

[159] M. Simonnet, C. Gentric, E. Olmos and N. Midoux. Experimental determination of the drag coefficient in a swarm of bubbles. *Chem. Eng. Sci.* **62** (2007), 858–866.

Appendix A

Supporting Information

A.1 Chapter 5 – Grid Convergence

To show mesh convergence, the bounded simulation with $P_c = P_d$ is repeated for three additional meshes, one coarser and two finer. Figure A.1 shows the time evolution of the gas hold-up inside in the simulation domain over the period of 2.5 s. From Fig. A.1, the time evolution of the gas hold-up does not exhibit a discernible difference as the grid is refined from 22 052 elements to 39 402 elements. Thus, the mesh with 22 052 elements is used for all of the simulations reported in Chapter 5.

A.2 Chapter 6 – Grid Convergence

Simulations of channel flow using a hyperbolic tangent diffuse-interface with $\epsilon = 0.01$ are performed with three different meshes. Figure A.2 shows the time evolution of the overall gas hold-up over 2.5 s for simulations with 9800, 18 666 and 32 090 mesh elements. From Fig. A.2, the time evolution of the overall gas hold-up remains unchanged as the mesh is refined from 18 666 elements to 32 090. Therefore, the channel flow simulations in Chapter 6 are performed with 18 666 mesh elements.

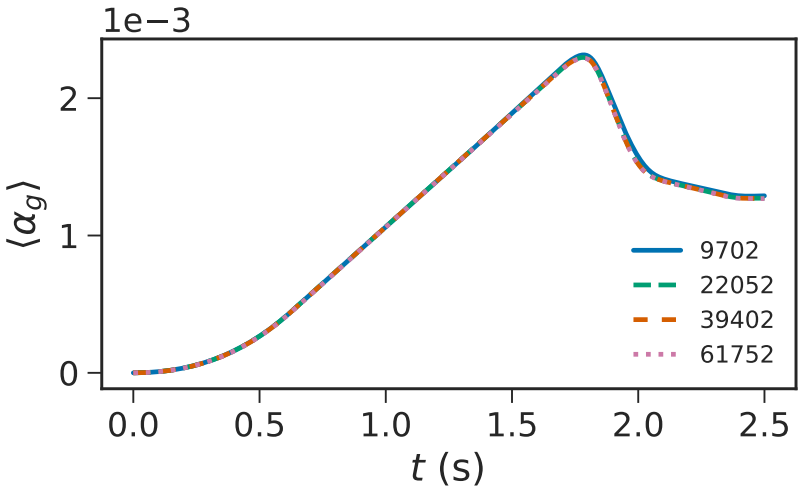


Figure A.1: Time evolution of gas hold-up for different number of mesh elements.

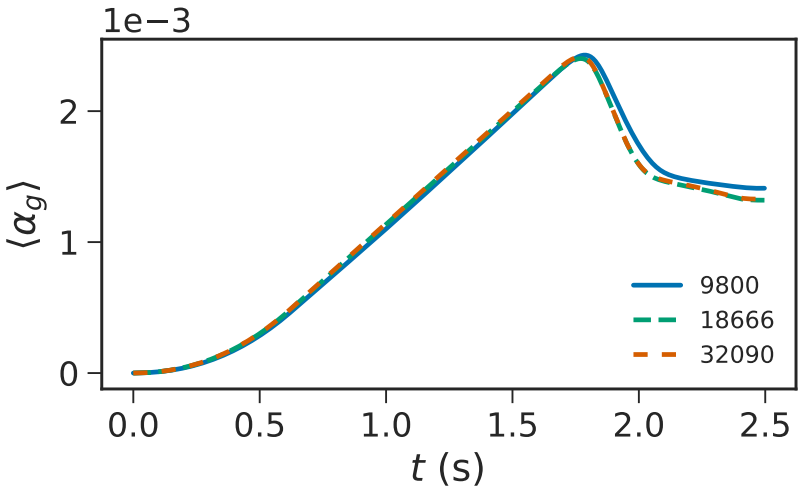


Figure A.2: Time evolution of gas hold-up obtained from the hyperbolic tangent diffuse-interface simulations of channel flow using different number of mesh elements.