

Balanced-force numerical framework for immiscible two phase flow at the onset of instability

Hrishikesh Joshi

Technische Universiteit Delft

BALANCED-FORCE NUMERICAL FRAMEWORK FOR IMMISCIBLE TWO PHASE FLOW AT THE ONSET OF INSTABILITY

by

Hrishikesh Joshi

in partial fulfillment of the requirements for the degree of

Master of Science
in Mechanical Engineering

at the Delft University of Technology,
to be defended publicly on Thursday September 28, 2017 at 02:00 PM.

Process and Energy Department Report number 2849

Thesis committee:

Prof. dr. ir. R. A. W. M. Henkes	TU Delft, supervisor
Dr. ir. D. R. van der Heul	TU Delft, supervisor
Dr. ir. W. P. Breugem	TU Delft
Prof. dr. ir. B. J. Boersma	TU Delft

An electronic version of this thesis is available at <http://repository.tudelft.nl/>.

"The most beautiful thing we can experience is the mysterious. It is the source of all true art and science."
-Albert Einstein

ACKNOWLEDGMENTS

Firstly, I would like to thank Dr.ir.D.R van der Heul for providing me an opportunity to work on this project and for the patience, the support and the encouragement he offered. I thank you for being extremely approachable and for providing critical feedback during the course of the thesis. I learnt from you the importance of retaining one's curiosity, being perpetually inquisitive and to dig deeper beneath the surface of the problem. I would also like to thank Prof.dr.ir.R.A.W.M Henkes for his direction, advice and the fruitful discussions we had. I thank you for proof reading my thesis and motivating me to keep on pushing forward. I hope I get to collaborate with the both of you in the near future.

I am greatly indebted to my parents and my brother, Aniket, for the unconditional support, love and belief they had in me.

This research has helped me gain a deeper understanding in the field of Numerical Analysis, and gave me a small insight into being a researcher. A further perk of working on this thesis was working alongside the best office mates one can ask for: Vineeth, Deborah, Merel and Hassan, who helped lighten the moments and made the journey worthwhile and enjoyable.

ABSTRACT

The Interface Capturing method, which is a finite volume method as formulated by Queutey and Vissonneau[1] for free surface immiscible, incompressible multiphase flows employs a collocated arrangement of unknowns and achieves a discrete force balance for the case when the interface coincides with the faces of the control volumes in the computational domain. This constraint limits the applicability of the method. Furthermore, the authors do not provide the exact formulation of the operators involved in the pressure velocity coupling.

In the present research, a balanced-force numerical method is formulated, applicable for an interface that neither has to coincide nor be aligned with the faces of the control volumes. The approach consists of the reconstruction of the values of the flow variables at the interface based on the interface jump conditions, with which the limit values of the normal derivatives at the interface are calculated. Furthermore, the construction of the operators of the discrete system is delineated to achieve a discrete force balance, by incorporating the reconstructed flow variables and employing a discretization which complies with the interface jump conditions. It is sufficient for a stationary discrete formulation to comply with the differential equation and the interface jump conditions. However, to apply this approach to solve unsteady flow problems the influence of the reformulated operators on the stability properties of the system should also be investigated.

The properties of the individual operators are analyzed as well as their behaviour when they are embedded in the complete solver algorithm. Results are shown for both steady and unsteady test cases and compared with numerical results obtained with OpenFOAM. The resulting framework avoids the occurrence of spurious velocities as it discretely complies with the interface conditions.

Keywords: Multiphase flows, immiscible, incompressible, balanced-force method, arbitrary interface, collocated arrangement of unknowns, Rhie and Chow pressure velocity coupling

CONTENTS

List of Figures	xi
1 Introduction	1
2 Physics of Two Phase flows	3
2.1 Flow Domain	3
2.2 Governing Equations	3
2.2.1 Conservation of Mass	4
2.2.2 Conservation of Momentum	4
2.3 Interface Conditions	5
2.3.1 Velocity Interface Conditions	5
2.3.2 Stress Interface Conditions.	5
2.4 Interface Jump Conditions	6
2.4.1 Matrix Form of interface conditions	6
2.4.2 Stress interface jump conditions.	6
2.4.3 Velocity interface jump conditions.	7
2.5 Conservative or Non-Conservative form	8
2.5.1 Conservative form	8
2.5.2 Non-conservative form	9
2.6 Overview	9
3 Interface Methods	11
3.1 Level Set Method	11
3.1.1 Time Evolution of level set function	11
3.1.2 Level Set Re-initialization	12
3.1.3 Mass Conservation.	12
3.2 Volume of Fluid Method	12
3.2.1 Time evolution of the Colour function	12
3.2.2 Discretization of Advection equation of colour function	13
3.2.3 Piecewise Linear Interface Calculation.	13
3.2.4 Compressive Method	14
3.2.5 Mass Conservation.	15
3.3 Comparison between Interface Methods	15
3.4 Information about the interface location	16
4 Pressure Velocity Coupling	17
4.1 Rhie and Chow Pressure-Velocity coupling	17
4.1.1 Interpretation of face Velocity	18
4.2 Iterative Solution Method	20
4.3 Open Questions.	21
4.4 Modifications in multiphase flow	22
4.4.1 Body force correction	22
4.4.2 Coefficient Correction	22
4.5 Overview	23
5 Discrete Force Balance	25
5.1 Gravity consistent discretization	25
5.1.1 Discrete Momentum Equation.	26
5.2 Modified Pressure.	27
5.2.1 Motivation	27
5.2.2 Mathematical Validity of Modified Momentum equation	27

5.3	Ghost Fluid Method.	27
5.3.1	Boundary Condition Capturing Ghost Fluid method.	28
5.4	Face value reconstruction based on second order Taylor expansion.	29
5.5	Comparison between Balanced-Force methods.	30
6	Numerical Challenges in modeling flow at the onset of instability	33
6.1	Continuous form of Governing Equations.	33
6.2	Discrete form of governing equations.	34
6.2.1	Discontinuities in flow fields.	34
6.2.2	Conservation of mass and momentum.	34
6.2.3	Discrete Force Balance.	34
7	Research Proposal	35
7.1	Research Objective	35
7.2	Research Outline	35
8	Balanced-Force discretization: Coinciding Interface and Cell Face	39
8.1	Interface jump conditions	39
8.1.1	Candidate test case	39
8.2	Quiescent Flow: Interface Coinciding with cell faces on a uniform grid	40
8.2.1	Pressure Reconstruction at the interface.	40
8.2.2	Boundary Conditions	41
8.2.3	Operator B	42
8.2.4	Operator DG^f	43
8.2.5	Result	44
8.3	Quiescent Flow: Interface Coinciding with the faces of the control volumes on a non-uniform grid	44
8.3.1	Pressure reconstruction at the interface	44
8.3.2	Discrete operators	45
8.3.3	Result	45
8.4	Discussion	46
9	Balanced-force discretization: Non coinciding Interface and Cell Face	47
9.1	General Formulation	47
9.2	Quiescent flow: Cell face offset and aligned with the interface on a non-uniform grid	48
9.2.1	Pressure reconstruction at the interface and the face	48
9.2.2	Operator DG^f	49
9.2.3	Operator B^c	50
9.2.4	Balanced-force approximation of the density	51
9.2.5	Result	52
9.3	Inclined interface	53
9.3.1	Pressure reconstruction at the face and the interface	53
9.3.2	Balanced-force approximation of the density	55
9.3.3	Discrete operators	57
9.3.4	Boundary Condition	58
9.3.5	Results	58
9.4	Method of Manufactured Solutions	58
9.4.1	Continuous form of Discrete operators	58
9.4.2	Convergence of discrete operators	61
9.4.3	Discussion	63
9.5	Velocity Reconstruction.	63
9.5.1	Reconstruction of the velocity at the interface	64
9.5.2	Reconstruction of velocity at the face	66
9.5.3	Discretization	66
9.5.4	Analytical Solution.	67
9.5.5	Results	67

10 Integrated Solver	69
10.1 Navier-Stokes Solver	69
10.2 Case: Closed Box	71
10.2.1 Quiescent Flow	72
10.2.2 Viscous wave.	75
10.2.3 Dam Break.	80
10.3 Stratified flow	82
10.3.1 Inflow-Outflow Boundary Conditions	82
10.3.2 Periodic Boundary Conditions	84
10.3.3 Analysis	85
11 Conclusions	89
A Appendix-a	91
A.1 Stencils of operator B	91
A.2 Stencil of operator DG^f	92
B Appendix-b	93
Bibliography	101

LIST OF FIGURES

2.1	Two-fluid stratified multiphase flow domain for analysis of governing equations	3
3.1	Example of 2D computational with coincident interface and mesh	13
3.2	Upwind, Donor, Acceptor arrangement of cells in Compressive schemes	14
4.1	Example of a 2D computational grid with collocated arrangement of unknowns	17
5.1	Example of a 2D computational grid with collocate arrangement of unknowns, with coincident interface and cell edge	25
5.2	Computational grid with adjacent cells L, R with coincident interface on face f	29
5.3	Computational grid with adjacent cells L, R with coinciding interface with the face f	30
7.1	Example of a coincing interface with the mesh	36
7.2	Example of an offset interface to the mesh	36
7.3	Example where the interface cuts two opposite faces of a cell	37
7.4	Example where the interface cuts two adjacent faces of the cell	37
8.1	2D Computational grid with the collocated arrangement of unknowns, in the case of coincident interface and cell edge	40
8.2	Implemented computational Grid with coincident interface and cell face	42
8.3	2D Computational non-uniform grid with collocated arrangement of unknowns	44
8.4	2D Computational non-uniform grid with collocate arrangement of unknowns	45
9.1	Example of a arbitrarily oriented and located interface	47
9.2	Example of offset interface to the face on a 2D collocated grid: discretization for Cell R	48
9.3	Example of offset interface to the face on a 2D collocated grid: discretization for Cell L	50
9.4	Example of an inclined interface : Face reconstruction from cell center below the interface	53
9.5	Example of an inclined interface: Face value reconstruction from cell center from above the interface	55
9.6	Example of an inclined interface: Face value reconstruction from cell center to the left of the interface	55
9.7	Example of an inclined interface: Face value reconstruction from cell center to the right of the interface	56
9.8	Computational Domain utilized in the Method of Manufactured Solutions	61
9.9	Order of the local truncation error in operator B near the interface	62
9.10	Order of the local truncation error in operator B away from the interface	62
9.11	Order of the local truncation error in operator DG^f near the interface	62
9.12	Order of the local truncation error in operator DG^f away from the interface	62
9.13	Velocity Recinstruction: Example of a offset and aligned interface	64
9.14	Stratified flow: Plot of numerical and analytical solution of velocity	67
9.15	Order of velocity reconstruction near the interface	67
10.1	Closed box case: Boundary conditions	71
10.2	Offset and aligned interface: Plot of L^∞ norm of error in u_1^f	73
10.3	Offset and aligned interface: Plot of L^∞ norm of error in u_2^f	73
10.4	Offset and aligned interface: Plot of L^∞ norm of error in u_1^c	73
10.5	Offset and aligned interface: Plot of L^∞ norm of error in u_2^c	73
10.6	Offset and aligned interface: Plot of initialized perturbation in pressure against x_2	74
10.7	Offset and aligned interface: Plot of L^∞ norm of error in pressure	74

10.8	Offset and aligned interface perturbed case: Plot of L^∞ norm of the error in u_1^f	74
10.9	Offset and aligned interface perturbed case: Plot of L^∞ norm of the error in u_2^f	74
10.10	Offset and aligned interface perturbed case: Plot of L^∞ norm of the error in u_1^c	75
10.11	Offset and aligned interface perturbed case: Plot of L^∞ norm of the error in u_2^c	75
10.12	Closed box: Initialized viscous standing wave	76
10.13	Viscous standing wave: Elevation plot for $Re = 2$	76
10.14	Viscous standing wave: Quiver plot at $t = 2.5s$ for the Implemented Solver	77
10.15	Viscous standing wave: Quiver plot at $t = 2.5s$ for OpenFoam	77
10.16	Viscous standing wave: Comparison of velocity at $x_1 = 0.5$	77
10.17	Viscous standing wave: Velocity plot at $x_2 = 0.5$ for the Implemented Solver	78
10.18	Viscous standing wave: Velocity plot at $x_2 = 0.5$ for OpenFoam	78
10.19	Mesh refinement: Velocity plot along the vertical line at $x_1 = 0.5$	78
10.20	Mesh refinement: Velocity plot along the vertical line at $x_2 = 0.5$	78
10.21	Viscous standing wave: Elevation plot at $Re = 20$	79
10.22	Viscous standing wave: Elevation plot at $Re = 200$	79
10.23	Dam Break: Quiver plot of velocity and contour of the level set field at $t=0.1s$	80
10.24	Dam Break: Quiver plot of velocity and contour of the level set field at $t=0.2s$	80
10.25	Dam Break: Quiver plot of velocity and contour of the level set field at $t=0.3s$	81
10.26	Dam Break: Quiver plot of velocity and contour of the level set field at $t=0.7s$	81
10.27	Dam Break: Contour of level set field: $t=0.3s$	81
10.28	Dam Break: Contour of level set field: $t=0.4s$	81
10.29	Dam Break: Contour of level set field: $t=0.7s$	81
10.30	Dam Break: Contour of level set field: $t=0.8s$	81
10.31	Stratified flow: Inflow-Outflow condition	82
10.32	Stratified flow: Periodic Boundary condition	82
10.33	Stratified flow: Plot of L^∞ norm of the error in u_f^1	83
10.34	Stratified flow: Plot of L^∞ norm of the error in u_2^f	83
10.35	Stratified flow: Plot of L^∞ norm of the error in u_1^c	83
10.36	Stratified flow: Plot of L^∞ norm of the error in u_2^c	83
10.37	Stratified flow Perturbed case: Plot of L^∞ norm of the error in u_f^1	84
10.38	Stratified flow Perturbed case: Plot of L^∞ norm of the error in u_2^f	84
10.39	Stratified flow Perturbed case: Plot of L^∞ norm of the error in u_1^c	84
10.40	Stratified flow Perturbed case: Plot of L^∞ norm of the error in u_2^c	84
10.41	Stratified flow Periodic boundaries: Plot of L^∞ norm of the error in u_f^1	85
10.42	Stratified flow Periodic boundaries: Plot of L^∞ norm of the error in u_2^f	85
10.43	Stratified flow Periodic boundaries: Plot of L^∞ norm of the error in u_1^c	85
10.44	Stratified flow Periodic boundaries: Plot of L^∞ norm of the error in u_2^c	85
A.1	Location of the control volumes: Operator B	92
A.2	Location of the control volumes: Operator DG^f	92

1

INTRODUCTION

Multiphase flows are ubiquitous in marine, chemical, oil and gas industry applications. Boiling in steam generators, slug flow in oil pipelines and particle laden flow are a few examples of multiphase flows. Multiphase flow encompasses a wide spectrum of flows having more than one phase and can be classified as liquid-liquid, liquid-gas or liquid-solid flows. Liquid-gas flows can be further divided into dispersed or separated flows. Dispersed flows consist of finite sized bubbles or particles in a continuous phase, while separated flows consist of two or more streams of fluids having a distinct interface between them [2]. These flows widely vary in the flow dynamics according to the phases present and the flow application due to a large range of velocity, time and length scales. As a result numerically modeling multiphase flows necessitates the use of different numerical approaches per application. In a two phase channel flow, different flow regimes are encountered with varying superficial velocities of the two fluids. When the flow velocities are low, the flow is stratified, as the fluid velocity increases, the high inertia of the fluids creates an instability and the flow becomes unstable and transitions to slug flow. Numerical modeling of these flows poses a major challenge. A small numerical error unphysically affects the flow produced. The current research will focus on numerical modeling of two-phase stratified channel flow at the onset of instability.

To ensure a physically consistent model of the flow, the discretization of the governing equations has to satisfy the following properties:

1. Discrete force-balance

A basic property of a fluid is that it flows under the action of a shear stress, i.e. a surface force attributed to viscosity. Along with the viscous force, gravity, a body force, acts on the fluid. Newton's second law dictates that the time rate of change of the momentum is directly proportional to the force applied, which has to be invariably satisfied in both: the continuous and the discrete form. In the special case of quiescent flow, the pressure at any point in the fluid is balanced by the weight of the fluid above it, which means the pressure gradient has to balance gravity. Achieving an exact balance between the forces acting on the fluids is a necessary condition to accurately model multiphase flows.

2. Discrete conservation of mass and momentum

Mass or energy cannot be created or destroyed, which dictates the conservation of mass and the momentum in fluid dynamics. Achieving a discrete conservation of mass and momentum is the second necessary condition while modeling multiphase flow at the onset of instability.

Meeting the above two requirements is no easy task, we face the following numerical challenges while doing so:

1. Pressure-Velocity Coupling

The computational domain is decomposed into control volumes in which the flow variables are arranged either in a staggered or a collocated fashion. Formulating the discrete system on a staggered grid is easier, but extending this formulation to a geometrically complex domains is not so trivial. On the other hand, adapting the collocated grid on a geometrically complex domain is simpler. However, the collocated arrangement leads to a weakly coupled velocity vector and pressure field. Hence an

explicit pressure-velocity coupling algorithm has to be implemented. The most common and widely used is the Rhie-Chow pressure velocity coupling in which a numerical face velocity is formulated. The quantitative effect of the formulation on the discrete conservation of mass and momentum for multiphase flows is unclear.

2. Formulation of governing equations

The governing equations can be formulated and discretized in the conservative or the non-conservative form, and with different sets of dependent variables (for example momentum as velocity multiplied by the density or momentum as a variable itself). Unphysical high velocity streaks in the lighter phase have been reported in numerical simulations of a multiphase flow utilizing a conservative formulation of equations. It may be attributed to artificial numerical momentum transfer between the two phases at the interface. Further analysis is required to draw definite conclusions.

3. Force Balance on a general setting

Queutey and Visonneau [1] achieved an exact force balance for the case when the interface coincides with the faces of control volumes in the computational domain. Their proposed method employs a second order Taylor expansion to calculate the values of a flow variable at the interface by employing the *a priori* known interface jump conditions. The method is not exact when the interface does not coincide with cell faces and requires a high grid resolution near the interface. In this thesis, a balanced-force numerical framework independent of the location and the orientation of the interface with respect to the cell faces will be proposed.

4. Interface representation

The interface between the fluids present in the computational domain is tracked by solving for a scalar field. The Level Set Method (LS) uses a signed distance function to the interface, and the Volume of Fluid method (VoF) uses a colour function, of which the volumetric average represents a volume fraction of a fluid in each control volume. LS and the Algebraic-VoF method do not conserve mass, while the Geometric-VoF needs a complex interface reconstruction in each time step. Hence, the interface tracking method itself incurs numerical errors in addition to those attributed to the discretization of the Navier-Stokes equations, and forms a separate topic of research. In this thesis the discretization of the Navier Stokes equations for two-phase flow will be analyzed and implemented to achieve a balanced-force formulation assuming the interface is known, i.e. the objective is purely an investigation into the spatial discretization and the construction of the discrete operators involved.

2

PHYSICS OF TWO PHASE FLOWS

Conservation of mass and momentum govern the dynamics of the fluids and form the basis of a numerical method for simulating flows. The continuous equations describing the conservation of mass and momentum can be formulated in a conservative or a non-conservative form. The variables involved in the continuous form may represent physical or purely mathematical quantities, which will be conserved when discretized. In a multiphase flow the type of formulation may significantly affect the numerical solution according to the type of quantity being conserved. Moreover, due to the presence of two fluids having different fluid properties, including a large difference in density, the momentum field has a C^{-1} type discontinuity. Thus, care needs to be taken when discretizing the Navier-Stokes equations. In this Chapter, first the governing equations are derived, followed by the interface jump conditions. Finally, the different conservative and non-conservative formulations of the governing equations are discussed.

2.1. FLOW DOMAIN

Consider a general two-fluid flow domain as shown in Figure 2.1. The flow domain is split into subdomains Ω_1 , Ω_2 having fluids 1 and 2 respectively. The fluid properties are labeled with the appropriate subscript indicating fluid 1 or 2.

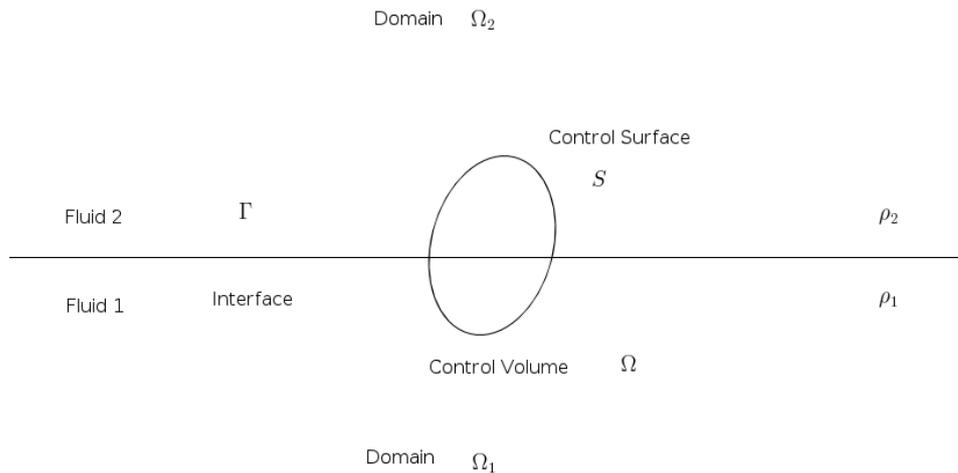


Figure 2.1: Two-fluid stratified multiphase flow domain for analysis of governing equations

2.2. GOVERNING EQUATIONS

Consider a control volume Ω , bounded by the control surface S that intersects the interface Γ as shown in Figure 2.1. Mass and momentum will be conserved for the flow in this control volume. The equations governing

conservation of mass and momentum equations will be derived in the following subsections.

REYNOLDS TRANSPORT THEOREM

Consider a material volume that moves with the flow such that the same fluid particles remain in the volume. Reynolds transport theorem states, that for a material volume Ω and a differentiable scalar ϕ

$$\frac{d}{dt} \int_{\Omega(t)} \phi d\Omega = \int_{\Omega(t)} \frac{d\phi}{dt} + \nabla \cdot (\phi \mathbf{U}) d\Omega. \quad (2.1)$$

The Reynolds transport theorem will be used to simplify the governing conservation equations.

2.2.1. CONSERVATION OF MASS

The mass in a control volume is conserved when the time rate of change of mass is equal to the mass source. Hence in the absence of a source term we have

$$\frac{d}{dt} \int_{\Omega(t)} \rho d\Omega = 0. \quad (2.2)$$

Applying the Reynolds Transport Theorem

$$\int_{\Omega(t)} \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{U}) d\Omega = 0. \quad (2.3)$$

Expanding the second term in (2.3), we get

$$\int_{\Omega(t)} \frac{\partial \rho}{\partial t} + \rho \nabla \cdot \mathbf{U} + \mathbf{U} \cdot \nabla \rho d\Omega = 0, \quad (2.4)$$

$$\int_{\Omega(t)} \frac{D\rho}{Dt} + \rho \nabla \cdot \mathbf{U} = 0. \quad (2.5)$$

The material derivative of density will be zero as the fluid is incompressible, hence

$$\int_{\Omega(t)} \nabla \cdot \mathbf{U} d\Omega = 0. \quad (2.6)$$

Equation (2.6) must hold for any $\Omega(t)$, hence

$$\nabla \cdot \mathbf{U} = 0. \quad (2.7)$$

2.2.2. CONSERVATION OF MOMENTUM

The momentum in a control volume is conserved when the rate of change of momentum is equal to the integrated surface and body forces acting on the fluid.

Consider a surface force f_s and a body force f_b acting on the fluid. The conservation of momentum is given by

$$\frac{d}{dt} \int_{\Omega(t)} \rho \mathbf{U} d\Omega = \int_{\Omega(t)} \rho \mathbf{f}_b d\Omega + \int_{S(t)} \mathbf{f}_s dS. \quad (2.8)$$

Applying the Reynold Transport Theorem

$$\int_{\Omega(t)} \frac{\partial \rho \mathbf{U}}{\partial t} + \nabla \cdot (\rho \mathbf{U} \mathbf{U}) d\Omega = \int_{\Omega(t)} \rho \mathbf{f}_b d\Omega + \int_{S(t)} \mathbf{f}_s dS. \quad (2.9)$$

The surface force consists of the viscous stress and pressure, hence assuming a Newtonian fluid

$$\mathbf{f}_s = -\nabla p + \mu(\nabla \mathbf{U} + \nabla \mathbf{U}^T). \quad (2.10)$$

Gravity acts as a body force on the fluid. Hence

$$\mathbf{f}_b = \mathbf{g}. \quad (2.11)$$

Substituting equations (2.11), (2.10) in (2.9), we get

$$\int_{\Omega(t)} \frac{\partial \rho \mathbf{U}}{\partial t} + \nabla \cdot (\rho \mathbf{U} \mathbf{U}) d\Omega = \int_{\Omega(t)} \rho \mathbf{g} d\Omega + \int_{S(t)} -\nabla p + \mu(\nabla \mathbf{U} + \nabla \mathbf{U}^T) dS. \quad (2.12)$$

As equation (2.12) should hold for any $\Omega(t)$, the integrand has to be zero, hence

$$\frac{\partial \rho \mathbf{U}}{\partial t} + \nabla \cdot (\rho \mathbf{U} \mathbf{U}) = -\nabla p + \rho \mathbf{g} + \mu(\nabla \mathbf{U} + \nabla \mathbf{U}^T). \quad (2.13)$$

DISCONTINUITY OF FLOW FIELDS IN MULTIPHASE FLOWS

Two phase flow consists of two fluids having different fluid properties in the computational domain, for example: $\rho(\mathbf{x}, t)$ is a step function with a discontinuity at the interface. As a result the term $\nabla \cdot (\rho \mathbf{U} \mathbf{U})$ has a jump discontinuity and is non differentiable at the interface and hence the partial differential equation is not mathematically valid on the interface. Consequently the flow domain is decomposed into two separate domains for the two fluids where the governing equations do hold. The solutions in the two separate domains are obtained by solving (2.13) and (2.7) in the two separate domains coupled by interface jump conditions.

2.3. INTERFACE CONDITIONS

In this section the interface jump conditions will be derived and discussed.

2.3.1. VELOCITY INTERFACE CONDITIONS

1. Kinematic condition

On the interface, between two immiscible fluids, mass conservation dictates that the velocity normal to the interface for both fluids must be equal, which is termed as the Kinematic Velocity Condition.

Let $\mathbf{U}_1, \mathbf{U}_2$ denote the velocity for the two fluids, \mathbf{N} the vector normal to the interface.

Hence the Kinematic Condition implies

$$\mathbf{U}_1 \cdot \mathbf{N} = \mathbf{U}_2 \cdot \mathbf{N}. \quad (2.14)$$

2. Dynamic condition

The interface in a multiphase flow is a surface between the two fluids, that moves with the flow. This implies that both fluids at the interface have equal velocities and hence have no slip. Mathematically it implies

$$\mathbf{U}_1 \cdot \mathbf{T} = \mathbf{U}_2 \cdot \mathbf{T}. \quad (2.15)$$

Here \mathbf{T} denotes the vector tangent to the interface.

Equations (2.14) and (2.15) imply that the velocity is continuous.

2.3.2. STRESS INTERFACE CONDITIONS

For the stress tensor $\overline{\overline{E}}$, a force balance on the massless interface dictates

$$\int \int_S (\overline{\overline{E}}_1 - \overline{\overline{E}}_2) \mathbf{N} = 0. \quad (2.16)$$

The total stress tensor is given by

$$\mathbf{E} = -p\mathbf{I} + \overline{\overline{\tau}}. \quad (2.17)$$

The deviatoric stress tensor $\overline{\overline{\tau}}$ is given by

$$\overline{\overline{\tau}} = \mu(\nabla \mathbf{U} + \nabla \mathbf{U}^T). \quad (2.18)$$

The velocity \mathbf{U} vector expressed in *Einstein summation notation* is

$$\mathbf{U} = u_i e_i.$$

The gradient of the velocity vector is a tensor, which using *Cartesian tensor notation*, is given by

$$\nabla \mathbf{U} = [u_{\alpha, \beta}]. \quad (2.19)$$

Differentiation using *Cartesian tensor notation* is denoted by

$$\frac{\partial \phi}{\partial x_\alpha} = \phi_{, \alpha}. \quad (2.20)$$

The *Cartesian tensor notation* and *Einstein summation convention* will be used in this thesis, and ∇ will be used wherever deemed more convenient than the tensor notation.

1. Tangential stress condition

Projecting the stress tensor in the direction tangent to the interface gives

$$\mathbf{T}^T (\bar{\bar{E}}_1 - \bar{\bar{E}}_2) \mathbf{N} = 0. \quad (2.21)$$

2. Normal stress condition

Similarly, projecting the stress tensor in the direction normal to the interface gives

$$\mathbf{N}^T (\bar{\bar{E}}_1 - \bar{\bar{E}}_2) \mathbf{N} = 0. \quad (2.22)$$

2.4. INTERFACE JUMP CONDITIONS

If the computational domain Ω is decomposed into two domains, Ω^+ , Ω^- for the two phases, then for a quantity Q , which is continuously differentiable away from the interface, an operator $[\]$ is defined, such that it represents the jump, i.e. the difference in Q over the interface. Hence

$$[Q] = Q^+ - Q^-, \quad (2.23)$$

where Q^+ represents the value in the limit of the interface in the subdomain Ω^+ and Q^- represents the value of in the limit of the interface in the subdomain Ω^- . With the help of the interface conditions specified in section 2.3, constraints can be derived that reveal the smoothness (order of continuity) of the flow variables. Kang et al. [3] derived these interface jump conditions in detail. These conditions have to be accounted for when discretizing the system of equations. The interface jump conditions will be described in the following sub-sections.

2.4.1. MATRIX FORM OF INTERFACE CONDITIONS

The stress conditions (2.21), (2.22) written in matrix form read:

$$\left[\begin{pmatrix} \mathbf{N}^T \\ \mathbf{T}_1^T \\ \mathbf{T}_2^T \end{pmatrix} (pI - \tau) \mathbf{N} \right] = \mathbf{0}, \quad (2.24)$$

where \mathbf{N} is the unit normal vector to the interface and \mathbf{T}_1 , \mathbf{T}_2 are orthogonal unit tangent vectors. Further, definition (2.18) for τ can be substituted in equation (2.24) to give

$$\begin{pmatrix} p \\ 0 \\ 0 \end{pmatrix} - \mu \begin{pmatrix} \mathbf{N}^T \\ \mathbf{T}_1^T \\ \mathbf{T}_2^T \end{pmatrix} \begin{pmatrix} \nabla u_1 \mathbf{N} \\ \nabla u_2 \mathbf{N} \\ \nabla u_3 \mathbf{N} \end{pmatrix} - \mu \begin{pmatrix} \nabla u_1 \mathbf{N} & \nabla u_2 \mathbf{N} & \nabla u_3 \mathbf{N} \\ \nabla u_1 \mathbf{T}_1 & \nabla u_2 \mathbf{T}_1 & \nabla u_3 \mathbf{T}_1 \\ \nabla u_1 \mathbf{T}_2 & \nabla u_2 \mathbf{T}_2 & \nabla u_3 \mathbf{T}_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}. \quad (2.25)$$

2.4.2. STRESS INTERFACE JUMP CONDITIONS

Equation (2.25) can be simplified to the following interface jump conditions:

$$[p - 2\mu ((\nabla u_1) \cdot \mathbf{N}, (\nabla u_2) \cdot \mathbf{N}, (\nabla u_3) \cdot \mathbf{N}) \cdot \mathbf{N}] = 0, \quad (2.26)$$

$$[\mu ((\nabla u_1) \cdot \mathbf{N}, (\nabla u_2) \cdot \mathbf{N}, (\nabla u_3) \cdot \mathbf{N}) \cdot \mathbf{T}_1 + ((\nabla u_1) \cdot \mathbf{T}_1, (\nabla u_2) \cdot \mathbf{T}_1, (\nabla u_3) \cdot \mathbf{T}_1) \cdot \mathbf{N}] = 0, \quad (2.27)$$

$$[\mu ((\nabla u_1) \cdot \mathbf{N}, (\nabla u_2) \cdot \mathbf{N}, (\nabla u_3) \cdot \mathbf{N}) \cdot \mathbf{T}_2 + ((\nabla u_1) \cdot \mathbf{T}_2, (\nabla u_2) \cdot \mathbf{T}_2, (\nabla u_3) \cdot \mathbf{T}_2) \cdot \mathbf{N}] = 0, \quad (2.28)$$

where the operator $[\]$ is defined in equation (2.23).

2.4.3. VELOCITY INTERFACE JUMP CONDITIONS

PROJECTION IN THE TANGENTIAL DIRECTION

As explained in Section 2.4 the velocity is continuous across the interface. Hence,

$$[\mathbf{U}] = 0. \quad (2.29)$$

Moreover, the tangential derivatives are continuous, hence

$$[\nabla \mathbf{U} \cdot \mathbf{T}] = 0, \quad (2.30)$$

where \mathbf{T} is any unit tangent vector to the interface.

Projecting Equation (2.30) in the tangential direction gives

$$[\nabla \mathbf{U} \cdot \mathbf{T} \cdot \mathbf{T}] = 0. \quad (2.31)$$

Equation (2.31) can be written as

$$\left[\frac{\partial \mathbf{U}_T}{\partial T} \right] = 0, \quad (2.32)$$

which states that the tangential derivative of the tangential velocity is continuous across the interface.

PROJECTION IN THE NORMAL DIRECTION

Similarly, taking the projection of equation (2.30) in the normal direction gives

$$[\nabla \mathbf{U} \cdot \mathbf{T} \cdot \mathbf{N}] = 0, \quad (2.33)$$

which can be written as

$$\left[\frac{\partial \mathbf{U}_T}{\partial N} \right] = 0. \quad (2.34)$$

This states that the normal derivative of the tangential velocity is continuous across the interface. Furthermore, identity (2.35) can be used to derive a jump condition for the normal derivative of the normal velocity:

$$(\nabla u_1 \cdot \mathbf{N}, \nabla u_2 \cdot \mathbf{N}, \nabla u_3 \cdot \mathbf{N}) \cdot \mathbf{N} + (\nabla u_1 \cdot \mathbf{T}_1, \nabla u_2 \cdot \mathbf{T}_1, \nabla u_3 \cdot \mathbf{T}_1) \cdot \mathbf{T}_1 + (\nabla u_1 \cdot \mathbf{T}_2, \nabla u_2 \cdot \mathbf{T}_2, \nabla u_3 \cdot \mathbf{T}_2) \cdot \mathbf{T}_2 = \nabla \cdot \mathbf{U} = 0. \quad (2.35)$$

Equation (2.35) can be re-written as

$$\left[\frac{\partial \mathbf{U}_n}{\partial n} + \frac{\partial \mathbf{U}_{T_1}}{\partial T_1} + \frac{\partial \mathbf{U}_{T_2}}{\partial T_2} \right] = 0. \quad (2.36)$$

The above equation can be simplified to

$$\left[\frac{\partial \mathbf{U}_n}{\partial n} \right] + \left[\frac{\partial \mathbf{U}_{T_1}}{\partial T_1} \right] + \left[\frac{\partial \mathbf{U}_{T_2}}{\partial T_2} \right] = 0, \quad (2.37)$$

as the tangential derivative of the tangential velocity is continuous across the interface. Hence

$$\left[\frac{\partial \mathbf{U}_N}{\partial N} \right] = 0, \quad (2.38)$$

which means the normal derivative of the normal velocity is continuous across the interface.

PROJECTION OF THE TANGENTIAL DERIVATIVE OF VELOCITY IN THE NORMAL DIRECTION

Further, equation (2.27) can be written as

$$\left[\mu \frac{\partial \mathbf{U}_N}{\partial T} + \mu \frac{\partial \mathbf{U}_T}{\partial N} \right] = 0, \quad (2.39)$$

where $\frac{\partial}{\partial T}$, $\frac{\partial}{\partial N}$ represent the tangential and normal directional derivative, respectively. This gives

$$\left[\mu \frac{\partial \mathbf{U}_N}{\partial T} \right] + \left[\mu \frac{\partial \mathbf{U}_T}{\partial N} \right] = 0. \quad (2.40)$$

As $\frac{\partial \mathbf{U}_T}{\partial N}$ is continuous across the interface from (2.34) it follows that

$$\begin{aligned} [\mu] \frac{\partial \mathbf{U}_T}{\partial N} + \left[\mu \frac{\partial \mathbf{U}_N}{\partial T} \right] &= 0, \\ \left[\mu \frac{\partial \mathbf{U}_N}{\partial T} \right] &= -[\mu] \frac{\partial \mathbf{U}_T}{\partial N}. \end{aligned} \quad (2.41)$$

SUMMARY OF INTERFACE JUMP CONDITIONS

For future reference the interface jump conditions that couple the solution for the two fluids are summarized below. These conditions have to be taken into account in the formulation of the discretization of (2.7) and (2.13) in two-phase flow.

$$\begin{aligned} [\mathbf{U}] &= 0, \\ \left[\frac{\partial \mathbf{U}_T}{\partial N} \right] &= 0, \\ \left[\frac{\partial \mathbf{U}_N}{\partial N} \right] &= 0, \\ \left[\frac{\partial \mathbf{U}_T}{\partial T} \right] &= 0, \\ \left[\mu \frac{\partial \mathbf{U}_N}{\partial T} \right] &= -[\mu] \frac{\partial \mathbf{U}_T}{\partial N}. \end{aligned} \quad (2.42)$$

2.5. CONSERVATIVE OR NON-CONSERVATIVE FORM

The governing equations to be satisfied by viscous incompressible flow are: the continuity equation, which dictates conservation of mass and the Navier-Stokes, equations which dictate conservation of momentum. The governing equations can be formulated in a conservative or a non-conservative form. As discussed previously, the terms in the two formulations may represent physical or mathematical quantities, which are conserved when discretized. Moreover different sets of dependent variables can be used to solve the governing equations, for example the mass flow rate can be expressed as $\rho \mathbf{U}$ or mass flow rate \mathbf{m} can be used as a variable itself. The choice of the formulation may affect the resulting numerical solution and needs to be further investigated.

In this section the different formulations of the continuous form of the equations are discussed.

2.5.1. CONSERVATIVE FORM

CONSERVATION OF MOMENTUM

The conservative form of the momentum equation can be written as,

$$\frac{\partial \rho \mathbf{U}}{\partial t} + \nabla \cdot (\rho \mathbf{U} \mathbf{U}) = -\nabla p + \nabla \cdot (\mu (\nabla \mathbf{U} + \nabla \mathbf{U}^T)) + \rho \mathbf{g}. \quad (2.43)$$

The gradient of pressure can be rewritten in divergence form by using the following identity,

$$\nabla \cdot (pI) = p(\nabla \cdot I) + I \cdot (\nabla p) = \nabla p, \quad (2.44)$$

as $\nabla \cdot I = 0$. Here I is the identity tensor.

Using the identity (2.44), equation (2.43) simplifies to

$$\boxed{\frac{\partial \rho \mathbf{U}}{\partial t} + \nabla \cdot (\rho \mathbf{U} \mathbf{U}) = \nabla \cdot (-p \mathbf{I} + \mu (\nabla \mathbf{U} + \nabla \mathbf{U}^T)) + \rho \mathbf{g}.} \quad (2.45)$$

$$\boxed{\frac{\partial \rho \mathbf{U}}{\partial t} + \nabla \cdot (\rho \mathbf{U} \mathbf{U} + p \mathbf{I} - \mu (\nabla \mathbf{U} + \nabla \mathbf{U}^T)) = \rho \mathbf{g}.} \quad (2.46)$$

Discretizing equation (2.46) will conserve momentum in the absence of external forces as the pressure, convective and viscous effects are taken into account in one divergence operator.

2.5.2. NON-CONSERVATIVE FORM

CONSERVATION OF MOMENTUM

The momentum equation in the non-conservative form is written as

$$\rho \frac{\partial \mathbf{U}}{\partial t} + \mathbf{U} \frac{\partial \rho}{\partial t} + \rho (\mathbf{U} \cdot \nabla) \mathbf{U} + \mathbf{U} \nabla \cdot (\rho \mathbf{U}) = -\nabla p + \nabla \cdot (\mu (\nabla \mathbf{U} + \nabla \mathbf{U}^T)) + \rho \mathbf{g}. \quad (2.47)$$

The equation can be further reduced with the continuity equation to give

$$\boxed{\rho \frac{\partial \mathbf{U}}{\partial t} + \rho (\mathbf{U} \cdot \nabla) \mathbf{U} = -\nabla p + \nabla \cdot (\mu (\nabla \mathbf{U} + \nabla \mathbf{U}^T)) + \rho \mathbf{g}.} \quad (2.48)$$

Identity (2.44) can be used to rewrite above equation as

$$\boxed{\rho \frac{\partial \mathbf{U}}{\partial t} + \rho (\mathbf{U} \cdot \nabla) \mathbf{U} = \nabla \cdot (-p \mathbf{I} + \mu (\nabla \mathbf{U} + \nabla \mathbf{U}^T)) + \rho \mathbf{g}.} \quad (2.49)$$

From conservation of mass we know \mathbf{U} is solenoidal (2.7), hence equation (2.48) can also be written as

$$\rho \frac{\partial \mathbf{U}}{\partial t} + \rho (\mathbf{U} \cdot \nabla) \mathbf{U} + \rho \mathbf{U} \nabla \cdot \mathbf{U} = -\nabla p + \nabla \cdot (\mu (\nabla \mathbf{U} + \nabla \mathbf{U}^T)) + \rho \mathbf{g}. \quad (2.50)$$

which simplifies to

$$\boxed{\rho \frac{\partial \mathbf{U}}{\partial t} + \rho \nabla \cdot (\mathbf{U} \mathbf{U}) = -\nabla p + \nabla \cdot (\mu (\nabla \mathbf{U} + \nabla \mathbf{U}^T)) + \rho \mathbf{g}.} \quad (2.51)$$

Equation (2.51) can also be rewritten with identity (2.44)

$$\boxed{\rho \frac{\partial \mathbf{U}}{\partial t} + \rho \nabla \cdot (\mathbf{U} \mathbf{U}) = \nabla \cdot (-p \mathbf{I} + \mu (\nabla \mathbf{U} + \nabla \mathbf{U}^T)) + \rho \mathbf{g}.} \quad (2.52)$$

Equation (2.52) is a very popular formulation of conservation of momentum used in the two-phase flow community. It is a non-conservative formulation of the conservation of momentum, albeit with the convective term in a conservative form.

Note: Different candidate formulations describing conservation of momentum have been boxed for easy reference.

2.6. OVERVIEW

A C^{-1} type discontinuity due to a density difference in the fluids creates discontinuous flow fields in a two-phase flow problem. Hence the derivatives of the flow fields do not exist at the interface. The governing equations can be solved away from the interface where the flow variables are continuous and hence exist. The dependence of the solution in the two subdomains is provided by the interface jump conditions.

The governing equations can be formulated in a conservative and non-conservative form. Moreover, the pressure can be formulated as a gradient and combined with the viscous term and the convective term can be rewritten in a conservative form. A different form of continuous equation leads to a different discretized formulation of the governing equation.

3

INTERFACE METHODS

The location and the orientation of the interface are required to incorporate the interface jump conditions. The location and the orientation are specified by the distance from a point to the interface and the normal vector to the interface, respectively. To this end an interface tracking method is employed which utilizes a scalar field to distinguish the fluids and locate the interface. Two of the most widely used methods are:

- The Level Set Method

The Level Set method uses a level set field: a scalar that represents the signed distance function from a point in the domain to the interface.

- The Volume of Fluid Method

The Volume of Fluid method utilizes a scalar: a colour function that has value 1 in one fluid and 0 in the other. Volume of fluid methods can be further classified as a:

- Geometric method
- Algebraic method

In this chapter, the Level Set method and the Volume of Fluid method are described, followed by a comparison and description of the advantages and disadvantages of each method.

3.1. LEVEL SET METHOD

The Level Set method employs an iso-contour of an implicit function ϕ , corresponding to value 0 to represent the interface. The scalar ϕ is positive in one fluid subdomain and negative in the other. An additional constraint

$$|\nabla\phi| = 1, \tag{3.1}$$

makes the implicit function a continuous signed distance function, such that for any position \mathbf{X} , we have

$$\phi = \begin{cases} 0, \mathbf{X} \in \Gamma, \\ d(\mathbf{X}, d\Gamma), |\nabla\phi| = 1, \mathbf{X} \in \Omega_1, \\ -d(\mathbf{X}, d\Gamma), |\nabla\phi| = 1, \mathbf{X} \in \Omega_2, \end{cases} \tag{3.2}$$

where Γ represents the interface, and Ω_1, Ω_2 are the domains occupied by fluids 1 and 2, respectively.

3.1.1. TIME EVOLUTION OF LEVEL SET FUNCTION

Consider a point \mathbf{X} on the interface Γ represented by the zero level set iso-contour. Hence

$$\frac{\partial\phi(\mathbf{X}(t), t)}{\partial t} = 0, \tag{3.3}$$

$$\frac{\partial \phi(\mathbf{X}, t)}{\partial t} + \frac{\partial \mathbf{X}(t)}{\partial t} \cdot \nabla \phi(\mathbf{X}, t) = 0. \quad (3.4)$$

As $\frac{\partial \mathbf{X}(t)}{\partial t} = \mathbf{U}$,

$$\frac{\partial \phi}{\partial t} + \mathbf{U} \cdot \nabla \phi = 0. \quad (3.5)$$

The time evolution of the level set function consists of advecting the scalar ϕ . Standard numerical methods can be used without the need for a complex interface reconstruction.

3.1.2. LEVEL SET RE-INITIALIZATION

The level set function loses the property of a signed distance function (magnitude of the gradient of the level set function should be equal to one) over the course of time steps. The magnitude of the gradient increases when the interface is stretched and reduces when it is compressed. The level-set function has to be re-initialized to re-establish the property of a signed distance function by solving

$$\begin{cases} \frac{\partial \phi}{\partial \tau} + |\nabla \phi| = 1, & \text{if } \mathbf{X} \in \Omega_1, \\ \frac{\partial \phi}{\partial \tau} - |\nabla \phi| = -1, & \text{if } \mathbf{X} \in \Omega_2. \end{cases} \quad (3.6)$$

Equations 3.6 are solved to steady state (for artificial time τ) to ensure constraint 3.1 is satisfied. This is referred to as the re-initialization of the level set function.

Above equations can also be written as

$$\frac{\partial \phi}{\partial t} = \text{sign}(\phi)(1 - |\nabla \phi|). \quad (3.7)$$

For more details reference to the work of Fedkiw, Osher [4] and Prosperetti [5] is made.

3.1.3. MASS CONSERVATION

The re-initialization of the level set function by standard numerical methods shifts the level set iso-contour corresponding to the interface, which leads to a loss of volume enclosed by the interface, i.e. a loss of mass.

3.2. VOLUME OF FLUID METHOD

The colour function, which is a Heavyside step function of ϕ , identifies the presence of a fluid at a particular location by

$$H(\phi) = \begin{cases} 1, & \text{if fluid 1} \\ 0, & \text{if fluid 2} \end{cases}. \quad (3.8)$$

Volumetric averaging of the colour function over a control volume gives the volume of fluid function, which represents the volume fraction of fluid 1 in that control volume. Hence,

$$C_i = \int_{\Omega_i} H(\phi) d\Omega. \quad (3.9)$$

3.2.1. TIME EVOLUTION OF THE COLOUR FUNCTION

The time evolution of the interface is given by

$$\frac{\partial C}{\partial t} + \mathbf{U} \cdot \nabla C = 0 \quad (3.10)$$

or, if the flow is incompressible, equivalently by

$$\frac{\partial C}{\partial t} + \nabla(\mathbf{U}C) = 0. \quad (3.11)$$

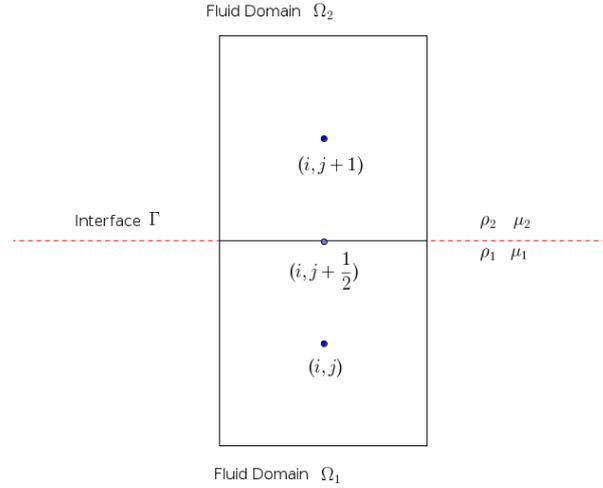


Figure 3.1: Example of 2D computational with coincident interface and mesh

VALIDITY OF THE VOLUME OF FLUID FUNCTION

Consider the case where the interface coincides with a cell edge as shown in the Figure 3.1. The interface is represented by the dashed red line. Cell $(i, j + 1)$ is occupied by fluid 2 and cell (i, j) by fluid 1. The scalar C forms a step profile with a discontinuity at $(i, j + \frac{1}{2})$. Hence, the gradient of the volume fraction function is undefined at $(i, j + \frac{1}{2})$. As a result, advecting equation 3.11 is mathematically valid only in a weak sense.

Most numerical frameworks employ a diffused interface formulation where the discontinuous fields are regularized. This permits definition of the gradient of the volume fraction function (among others). Ideally the discrete approximations in the finite volume method should respect all the physical constraints, keep the interface sharp and provide a solution without the need for regularization.

3.2.2. DISCRETIZATION OF ADVECTION EQUATION OF COLOUR FUNCTION

The Volume of Fluid methods are further classified as:

1. Geometric Methods

The surface is explicitly reconstructed and advected based on the geometric considerations. The simplest is the Simple Line Interface Calculation (SLIC) by Hirt and Nichols [6] where the interface in each cell is represented by two orthogonal lines parallel to the faces of the cell in a Cartesian mesh. Another widely used method, which is more accurate than SLIC, is the Piecewise Linear Interface Calculation (PLIC) by DeBar and Young [7]. PLIC involves representing the interface with a line (plane) of a constant slope in 2D (3D).

2. Algebraic Methods

Algebraic methods use special algebraic schemes that satisfy the local boundedness criteria and keep the interface sharp without smearing and without introducing numerical oscillations [8], [9]. The algebraic schemes utilize a compressive scheme when the flow is normal to the interface which steepens any gradient, and a high resolution scheme that maintains the sharpness of the interface when the flow is tangential to the interface. As an example the Compressive Interface Convection Scheme for Arbitrary Meshes (CICSAM) by Ubbink and Issa [10] is discussed in section 3.2.

3.2.3. PIECEWISE LINEAR INTERFACE CALCULATION

PLIC is a geometric VOF method which involves a linear reconstruction of the interface. The method of reconstruction and the time evolution are discussed in the following subsections.

INTERFACE RECONSTRUCTION

PLIC represents the interface by a line in 2D with a constant slope given by

$$m_x x + m_y y = \alpha. \quad (3.12)$$

The surface reconstruction consists of solving for the slope m and the constant α , after which the interface can be advected by geometric considerations.

Surface reconstruction can be done in many ways. A simple example is Young's finite difference method [7], where the slope is calculated as the gradient of the colour function. Hence

$$m = -\nabla C. \quad (3.13)$$

The above approach can be employed if the colour function is continuous in the computational domain. After calculating the slope, α is obtained such that the area (volume in 3D) under the line (surface in 3D) is equal to the volume fraction of the fluid in that cell.

ADVECTION OF THE INTERFACE

To advect the interface, the end points of the line segment in the cell are calculated. The end points are advected by the required time step along one co-ordinate direction by linearly interpolating the velocity, followed by reconstruction of the interface. This step is repeated in the other coordinate directions.

Hence, consider an end point x , which is advected along the x axis as,

$$x^{n+1} = x^n + \Delta t u(x^n). \quad (3.14)$$

where the interpolated velocity $u_1(x^n)$ is calculated as

$$u_1(x^n) = u_{1\ i-\frac{1}{2},j}(1-x) + u_{1\ i+\frac{1}{2},j}x. \quad (3.15)$$

This method is called *out-of-cell explicit linear mapping*, in contrast to this, onto-cell implicit linear mapping uses the velocity at the current time level (u_1^{n+1}).

The interested reader is referred to [7]

3.2.4. COMPRESSIVE METHOD

Compressive schemes such as CICSAM [10] convect the colour function by blending two high-resolution schemes. As an example of this approach, Ubbink's and Issa's seminal work on Compressive Interface Capturing Scheme for Arbitrary Meshes (CICSAM) will be discussed.

In a compressive scheme, the method is formulated based on the values at the center cell (Donor cell), the cell that receives fluid (Acceptor cell) and the upwind cell, labeled as D, U, A, respectively as shown in Figure 3.2.

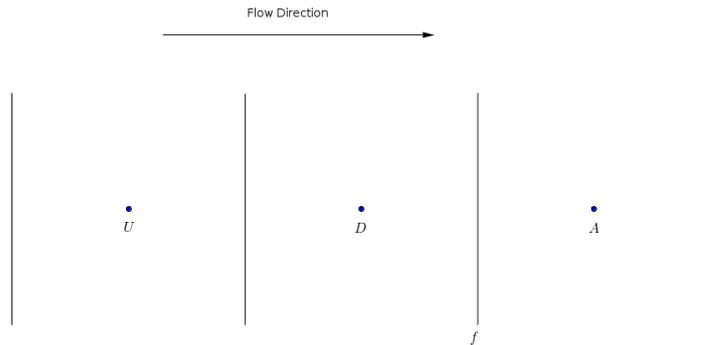


Figure 3.2: Upwind, Donor, Acceptor arrangement of cells in Compressive schemes

The original VoF scheme proposed by Hirt and Nichols [6] used an upwind scheme when the angle between the interface and flow direction was between $0^\circ - 45^\circ$ and downwind when it was between $45^\circ - 90^\circ$. Downwinding compresses any finite gradient to a step profile, making it suitable when the flow is normal to the interface. Upwinding satisfies the local boundedness criteria but smears the interface and downwinding maintains resolution but violates local boundedness [8] [10]. The original VoF method unphysically deforms the interface due to unboundedness of downwinding and the sudden shift between the two schemes.

A numerical formulation is needed which gradually changes from a compressive scheme (giving a step profile for finite gradient) when the flow is normal to the interface to one that maintains the resolution of the interface when the flow is tangential to the interface, while being locally bounded.

Ubbink and Issa formulated a blending formulation, weighted by the angle between the interface and the motion of the flow, of UTMATE-QUICKEST(UQ) and HYPER-C (upper bound of CBC) [10].

The schemes are formulated in terms of normalized variables defined as follows:

$$\tilde{f} = \frac{f - f_u}{f_a - f_u}, \quad (3.16)$$

f_u, f_a are the values at the upwind and the acceptor cell, respectively.

The upper bound of the CBC for explicit implementation is given by

$$\tilde{f}_f^{CBC} = \begin{cases} \min\left(\frac{\tilde{f}_D}{c}, 1\right) & \text{when } 0 \leq \tilde{f}_D \leq 1 \\ \tilde{f}_D & \text{when } \tilde{f}_D < 0, \tilde{f}_D > 0 \end{cases}. \quad (3.17)$$

The upper bound of the UQ for explicit implementation is given by

$$\tilde{f}_f^{UQ} = \begin{cases} \min\left(\frac{8c\tilde{f}_D + (1-c)(6\tilde{f}_D + 3)}{8}, \tilde{f}_f^{CBC}\right) & \text{when } 0 \leq \tilde{f}_D \leq 1 \\ \tilde{f}_D & \text{when } \tilde{f}_D < 0, \tilde{f}_D > 0 \end{cases}, \quad (3.18)$$

where c is the Courant number.

The value at the face is calculated by blending the upper-bound of HYPER-C and QUICKEST as

$$\tilde{f}_f = \gamma_f \tilde{f}_f^{CBC} + (1 - \gamma_f) \tilde{f}_f^{UQ}. \quad (3.19)$$

where,

$$\theta_f = \left| \frac{\nabla f_D \cdot d_f}{|\nabla f_D| |d_f|} \right|, \quad (3.20)$$

$$\gamma_f = \min\left(\kappa \frac{\cos(2\theta_f + 1)}{2}, 1\right). \quad (3.21)$$

κ determines the dominance of the different methods.

3.2.5. MASS CONSERVATION

Volume of fluid methods discretely conserve mass.

3.3. COMPARISON BETWEEN INTERFACE METHODS

An ideal interface method should conserve mass, should give an accurate advection and should not necessitate a computationally intensive interface reconstruction. Formulating such method is not trivial and is open to further research.

The advantages and disadvantages of the discussed interface tracking methods have been summarized in the table below.

Method	Advantages	Disadvantages
Level Set Method	-Advection of a signed distance function can be done with standard numerical methods. -No surface reconstruction needed.	-Level Set reinitialization leads to mass loss.
VoF-Geometric	-Exact conservation of mass on discrete level achieved.	-Complex surface reconstruction needed. -Non-unique interface produced depending on Geometric method used.
VoF- Algebraic	-No surface reconstruction needed.	-Does not conserve mass discretely. -Small time step needed, as CFL number restricted to a small value to maintain sharpness of the interface.

3.4. INFORMATION ABOUT THE INTERFACE LOCATION

Numerical frameworks for multiphase flows require the interface location and orientation to incorporate the interface jump conditions. The scalar field in the LS method represents a signed distance function, which can then be trivially employed to extract the required information. Geometric VoF methods provide this information, as the interface reconstruction is an integral part of the method. Whereas, Algebraic-VoF fail to provide any data at all. As a result, for the present research the choice of the interface tracking method is limited to the Level Set and the Geometric-VoF method.

4

PRESSURE VELOCITY COUPLING

It is relatively straightforward to discretize the Navier-Stokes equations on a geometrically complex domain employing a collocated arrangement of unknowns in comparison with a staggered arrangement of the unknowns. Hence, the collocated grid is commonly the preferred choice. However it is well known that the discretization on a collocated grid, without special measures, leads to a weakly coupled pressure and velocity vector field. Therefore an additional pressure velocity coupling algorithm needs to be implemented to re-establish the coupling between the two fields.

In this chapter first the Rhie and Chow pressure velocity coupling for single phase flow is discussed, followed by an overview of the necessary modifications to apply this approach to multiphase flows.

4.1. RHIE AND CHOW PRESSURE-VELOCITY COUPLING

Consider the 2D Cartesian grid shown in Figure 4.1. P is the control volume under consideration, E is the control volume to the east and N to the north of P .

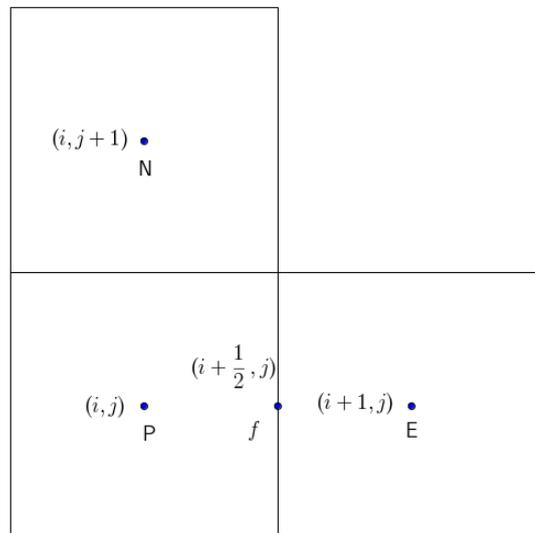


Figure 4.1: Example of a 2D computational grid with collocated arrangement of unknowns

The momentum equation in the $\hat{\mathbf{e}}_1$ direction in a discretized form for control volume P can symbolically be written as

$$\frac{\rho|\Omega_P|}{\Delta t}(u_P - u_P^n) + a_P u_P - \sum_P a_i u_i = -p_{,1} \Big|_P |\Omega_P|, \quad (4.1)$$

where P is the cell center over which momentum conservation is being applied, \sum_P represents the summation of terms corresponding to the surrounding cells (represented by index i) of cell P , Ω_P is the cell volume and n the current time level. The coefficients a_p in the summation represent the coefficients arising out of discretization of the viscous and the convective term in the control volume P and a_i corresponding to the coefficients in the discretization stencil of the surrounding control volumes. The time level at which other terms are evaluated is determined by the employed method of time integration.

Equation (4.1) simplifies to

$$\left[1 + \frac{\rho|\Omega_P|}{a_P\Delta t}\right] u_P = \frac{1}{a_P} \sum_P a_i u_i - \frac{|\Omega_P|}{a_P} p_{,1}|_P + \frac{\rho|\Omega_P|}{a_P\Delta t} u_P^n. \quad (4.2)$$

Similarly, the discrete form of the governing conservation of momentum equation evaluated over control volume E in $\hat{\mathbf{e}}_1$ is given by

$$\left[1 + \frac{\rho|\Omega_E|}{a_E\Delta t}\right] u_E = \frac{1}{a_E} \sum_E a_i u_i - \frac{|\Omega_E|}{a_E} p_{,1}|_E + \frac{\rho|\Omega_E|}{a_E\Delta t} u_E^n. \quad (4.3)$$

The Rhie and Chow Coupling or Pressure Weighted Interpolation Method (PWIM) [11] [12] constructs a face velocity u_f analogous to the (4.2) as

$$\left[1 + \frac{\rho|\Omega_f|}{a_f\Delta t}\right] u_f = \frac{1}{a_f} \sum_f a_i u_i - \frac{|\Omega_f|}{a_f} p_{,1}|_f + \frac{\rho|\Omega_f|}{a_f\Delta t} u_f^n. \quad (4.4)$$

The coefficients a_f and $\sum_f a_i$ are artificial numerical constructs. They are eliminated employing the following definitions:

$$\frac{1}{a_f} \sum_f a_i u_i = \frac{1}{2} \left(\frac{1}{a_P} \sum_P a_i u_i + \frac{1}{a_E} \sum_E a_i u_i \right) \quad (4.5)$$

$$\frac{|\Omega_f|}{a_f} = \frac{1}{2} \left(\frac{|\Omega_P|}{a_P} + \frac{|\Omega_E|}{a_E} \right) \quad (4.6)$$

Utilizing (4.2), (4.3) and (4.5),(4.6) the expression for the face velocity simplifies to

$$u_f = \frac{u_P + u_E}{2} - \left[\frac{1}{2} \left(\frac{|\Omega_P|}{a_P} + \frac{|\Omega_E|}{a_E} \right) p_{,1}|_f - \left(\frac{|\Omega_P|}{a_P} p_{,1}|_P + \frac{|\Omega_E|}{a_E} p_{,1}|_E \right) \right] - \left[\left(\frac{\rho|\Omega_P|}{a_P\Delta t} u_P^n + \frac{\rho|\Omega_E|}{a_E\Delta t} u_E^n \right) - \frac{1}{2} \left(\frac{\rho|\Omega_P|}{a_P\Delta t} + \frac{\rho|\Omega_E|}{a_E\Delta t} \right) u_f^n \right]. \quad (4.7)$$

The mass is discretely conserved over a control volume by imposing a solenoidality constraint on the derived Rhie and Chow face velocity field u_f .

4.1.1. INTERPRETATION OF FACE VELOCITY

The derived PWIM face velocity does not have any explicitly clear physical basis but is numerically formulated and simplified using the definitions (4.5) and (4.6). The justification for the formulation of the face velocity is presented by Miller and Schmidt [13]. Furthermore a physical rationalization can be found in the work of Miller and Schmidt [13] and Francois et al. [14].

In the following section, the interpretations of the numerically constructed face velocity are discussed to gain an intuitive understanding of PWIM.

SECOND-ORDER TAYLOR EXPANSION

The velocities at the cell centers P and E can be expressed in terms of the velocity at the face by retaining a second order Taylor expansion as

$$u_P = u_f - \Delta x u_{,1}|_f + \frac{\Delta x^2}{2!} u_{,11}|_f - \frac{\Delta x^3}{3!} u_{,111}|_f + \dots, \quad (4.8)$$

$$u_E = u_f + \Delta x u_{,1}|_f + \frac{\Delta x^2}{2!} u_{,11}|_f + \frac{\Delta x^3}{3!} u_{,111}|_f + \dots \quad (4.9)$$

Adding equations (4.8) and (4.9) and simplifying, gives

$$u_f = \frac{u_P + u_E}{2} - \frac{\Delta x^2}{8} u_{,11} \Big|_f + \mathcal{O}(\Delta x^4). \quad (4.10)$$

Application of a second order finite difference approximation to the second derivative gives

$$u_f = \frac{u_P + u_E}{2} - \frac{1}{2} (u_E + u_P - 2u_f) + \mathcal{O}(h^2). \quad (4.11)$$

Substituting (4.2), (4.3), (4.4) for the velocity in the second term in the parentheses and utilizing the definitions (4.5), (4.6) in equation (4.11) gives

$$u_f = \frac{u_P + u_E}{2} - \left[\frac{1}{2} \left(\frac{\Omega_P}{a_P} + \frac{\Omega_E}{a_E} \right) p_{,1} \Big|_f - \left(\frac{\Omega_P}{a_P} p_{,1} \Big|_P + \frac{\Omega_E}{a_E} p_{,1} \Big|_E \right) \right] - \left[\left(\frac{\rho \Omega_P}{a_P \Delta t} u_P^n + \frac{\rho \Omega_E}{a_E \Delta t} u_E^n \right) - \frac{1}{2} \left(\frac{\rho \Omega_P}{a_P \Delta t} + \frac{\rho \Omega_E}{a_E \Delta t} \right) u_f^n \right]. \quad (4.12)$$

INTERPOLATING VECTOR FIELD

Francois et al. [14] interpreted it as interpolating the whole vector field at the face rather than the velocity vector field.

Interpolating the velocity vector field at $(i + \frac{1}{2}, j)$ gives

$$\tilde{u}_f = \frac{u_E + u_P}{2}, \quad (4.13)$$

which, by using (4.2), (4.3) (ignoring terms at previous time level n for simplicity), results into

$$\tilde{u}_f = \frac{1}{2} \left(\frac{1}{\tilde{a}_P} \sum_P a_i u_i + \frac{1}{\tilde{a}_E} \sum_E a_i u_i \right) - \frac{1}{2} \left(\frac{\Omega_P}{\tilde{a}_P} p_{,1} \Big|_P + \frac{\Omega_E}{\tilde{a}_E} p_{,1} \Big|_E \right), \quad (4.14)$$

where $\tilde{a}_P = \frac{a_P}{1 + \frac{\rho \Omega_P}{a_P \Delta t}}$.

Such a discrete formulation on a collocated grid leads to a weak coupling between the velocity vector and the pressure field. Hence Rhie and Chow pressure velocity coupling formulates a pressure weighted interpolation. The face velocity is a function of the pressure at the face rather than the cell centers which leads to a stronger coupling between the pressure and the velocity field.

$$u_f = \tilde{u}_f + \frac{1}{2} \left(\frac{\Omega_P}{\tilde{a}_P} p_{,1} \Big|_P + \frac{\Omega_E}{\tilde{a}_E} p_{,1} \Big|_E \right) - \frac{\Omega_f}{\tilde{a}_f} p_{,1} \Big|_f \quad (4.15)$$

$$u_f = \frac{u_P + u_E}{2} + \frac{1}{2} \left(\frac{\Omega_P}{\tilde{a}_P} p_{,1} \Big|_P + \frac{\Omega_E}{\tilde{a}_E} p_{,1} \Big|_E \right) - \frac{\Omega_f}{\tilde{a}_f} p_{,1} \Big|_f \quad (4.16)$$

For more details refer to [14], [15].

A symbolic representation of the operators is employed for better readability and to concisely describe the formulations proposed in this thesis. This symbolic representation for the local operators, is described in the following subsection.

SYMBOLIC REPRESENTATION

For the sake of easy reference, the face velocity formulation will be described in an algebraic-differential form as

$$u_{\alpha \mathbf{i} \pm \hat{\mathbf{e}}_\alpha}^f = \bar{u}_{\alpha \mathbf{i}}^* + \frac{1}{\rho} \dot{B} p^{n+1}, \quad (4.17)$$

where

$$\bar{u}_{\alpha \mathbf{i}}^* = \frac{u_{\alpha \mathbf{i}}^* + u_{\alpha \mathbf{i} \pm \hat{\mathbf{e}}_\alpha}^*}{2}, \quad (4.18)$$

$$(\dot{B} p)_{\alpha \mathbf{i}}^\pm = \left[\frac{1}{2} \left(d_{\alpha}^h p_{\mathbf{i}} + d_{\alpha}^h p_{\mathbf{i} \pm \hat{\mathbf{e}}_\alpha} \right) - d_{\alpha}^{\frac{h}{2}} p_{\mathbf{i} \pm \frac{1}{2} \hat{\mathbf{e}}_\alpha} \right]. \quad (4.19)$$

The \dot{B}^+ operator refers to the face reconstruction at $\mathbf{i} + \frac{1}{2}\hat{\mathbf{e}}_\alpha$ and \dot{B}^- at $\mathbf{i} - \frac{1}{2}\hat{\mathbf{e}}_\alpha$ and the discrete operator d_α^h evaluates the α component of the pressure derivative.

The \dot{B} operator is further split as

$$\dot{B} = \dot{B}^c + \dot{B}^f, \quad (4.20)$$

where

$$\dot{B}^c = \frac{1}{2} \left(d_\alpha^h p_{\mathbf{i}} + d_\alpha^h p_{\mathbf{i} \pm \frac{1}{2}\hat{\mathbf{e}}_\alpha} \right), \quad (4.21)$$

$$\dot{B}^f = -d_\alpha^{\frac{h}{2}} p_{\mathbf{i} \pm \frac{1}{2}\hat{\mathbf{e}}_\alpha}. \quad (4.22)$$

Furthermore, the gradient operator at the face and at the center of the control volume is defined as

$$G^c = d_1^h p_{\mathbf{i}} + d_1^h p_{\mathbf{i}}, \quad (4.23)$$

$$G^{f \pm} = d_\alpha^{h/2} p_{\mathbf{i} \pm \frac{1}{2}\hat{\mathbf{e}}_\alpha}. \quad (4.24)$$

Hence, G^c represents the gradient of the pressure at the cell center, and $G^{f \pm}$ is utilized to represent the $\hat{\mathbf{e}}_\alpha$ component of the pressure gradient at the face having the index $\mathbf{i} \pm \frac{1}{2}\hat{\mathbf{e}}_\alpha$.

Moreover, the divergence of the operators needs to be calculated in the iterative approach of the pressure velocity coupling. Hence, for easy reference they are symbolically represented as

$$D\dot{B} = D\dot{B}^c + D\dot{B}^f, \quad (4.25)$$

where, D is the divergence operator. From (4.22) and (4.24) it can be seen that

$$D\dot{B} = B^f = DG^f. \quad (4.26)$$

Furthermore, the divergence of the cell center component of the operator \dot{B} is represented as

$$D\dot{B}^c = B \quad (4.27)$$

4.2. ITERATIVE SOLUTION METHOD

Miller and Schmidt [13] described an iterative solver for stationary flow based on the Rhie and Chow pressure-velocity coupling. The solution method consists of iteratively solving for the pressure and the velocity in two loops. The B and the DG^f operator are constructed in the outer loop, and the velocities and the pressure are updated in the inner loop. The operators are updated in the outer loop once the inner loop converges (convection part of the operators needs to be updated by linearizing the convection operator using the converged solution of the inner loop). The method has been described below in a differential-algebraic form.

start of the outer loop

(iteration level k)

1. Construct the diffusion operator J
2. Construct operators DG^f , B^c , G^c , G^f and the convective operator N

start of the inner loop

(iteration level l)

1. The momentum equations are solved implicitly to calculate the predicted cell center velocities \mathbf{u}_1^{c*} , \mathbf{u}_2^{c*} . The pressure gradient is calculated using the previous outer loop pressure values. The predicted cell center values are under-relaxed with respect to the outer loop cell center velocities.

$$\mathbf{u}^{c*} = (1 - \omega)\mathbf{u}^{c^k} + \omega \left(J\mathbf{u}^{c*} + N\mathbf{u}^* - G^c \mathbf{p}^k \right), \quad (4.28)$$

where G^c is the pressure gradient operator evaluated at the center of the control volume.

2. The predicted face velocities are calculated based on the predicted cell center velocities $\mathbf{u}_1^{f*}, \mathbf{u}_2^{f*}$.

$$\mathbf{u}^{f*} = (1 - \omega) (\mathbf{u}^{fk} - \tilde{\mathbf{u}}^{ck}) + \tilde{\mathbf{u}}^{c*} + \omega \hat{B} \mathbf{p}^l, \quad (4.29)$$

where $\tilde{\cdot}$ represents the interpolation of the flow variables based on the values at the adjacent cell centers.

3. The correction in the pressure δp is calculated by imposing the solenoidality condition on the corrected face velocities

$$\mathbf{u}^{f^{l+1}} = (1 - \omega) (\mathbf{u}^{fk} - \tilde{\mathbf{u}}^{ck}) + \tilde{\mathbf{u}}^{c^{l+1}} + \omega \hat{B} \mathbf{p}^{l+1} \quad (4.30)$$

Applying the solenoidality condition to (4.30), the final pressure correction equation becomes

$$DG^f \delta \mathbf{p} = D\mathbf{u}^{f*}, \quad (4.31)$$

where D is the divergence operator and G^f is the gradient operator to be applied to calculate the pressure gradients at the face.

4. The corrected cell center velocities are calculated $\mathbf{u}_1^{c^{l+1}}, \mathbf{u}_2^{c^{l+1}}$

$$u_\alpha^{c^{l+1}} = u_\alpha^{c*} - \frac{\omega}{1 - \omega} \frac{\Delta t}{\rho} \delta p_{,\alpha} \Big|_c \quad (4.32)$$

5. The corrected face velocities are calculated $u_1^{f^{l+1}}, u_1^{f^{l+1}}$

$$u_\alpha^{f^{l+1}} = u_\alpha^{f*} - \frac{\omega}{1 - \omega} \frac{\Delta t}{\rho} \delta p_{,\alpha} \Big|_f \quad (4.33)$$

6. The pressure is updated

$$\mathbf{p}^{l+1} = \mathbf{p}^l + \delta \mathbf{p} \quad (4.34)$$

end of the inner loop

The flow variables are updated to the values obtained in the last iteration level in the inner loop.

end of the outer loop

The loops are iteratively solved until convergence is obtained. The interested reader is referred to [13], [11].

4.3. OPEN QUESTIONS

The Rhie and Chow interpolation is a widely utilized pressure-velocity coupling method. As seen in section 4.1 the method consists of constructing a face velocity and applying a simplification employing definitions (4.5), (4.6). The constructed analogous equation for face velocity (4.4) has no physical basis but a rationalized one (section 4.1.1). Hence, the Rhie and Chow interpolation raises a few open questions concerning the extension of the method to two phase flows, as discussed below.

- **Conservation of momentum for face centered velocities**

The face velocity used to impose the conservation of mass has no defined control volume over which it satisfies the conservation of momentum. Hence writing an analogous equation for the face velocity has no physical basis, and conservation of momentum for the face velocity is unclear.

- **Conservation of mass for cell centered velocities**

The velocities that satisfy the conservation of momentum (the cell center velocities) have no explicit conservation of mass imposed, i.e the conservation of mass imposed on a control volume is not through the cell center velocities but through the constructed PWIM face velocities.

- **Linear variation**

Equations (4.5), (4.6) employ a linear interpolation, which implies quantities such as the pressure gradient vary linearly over the computational domain that categorically may not be true. Choi et al. [16] showed that if a coarse grid is employed or if the flow has a high pressure gradient it leads to an unphysical flow velocities as in such a case linearity cannot be assumed (violating assumptions (4.5), (4.6)). Similar conclusions were drawn by Miller and Schmidt [13], where a sudden constriction in a pipe flow produced a high axial pressure gradient which produces high unphysical velocities.

4.4. MODIFICATIONS IN MULTIPHASE FLOW

4.4.1. BODY FORCE CORRECTION

The Rhie and Chow pressure velocity coupling has to be modified in the presence of a body force as an incorrect treatment leads to an unphysical flow solution. The Rhie and Chow PWIM can be interpreted as calculating the terms (like the pressure gradient) at the face itself as opposed to a linear interpolation. In the presence of a body force which varies in the flow domain (an example being the buoyancy term, which is proportional to density) a similar treatment for the body force term was recommended by Gu [16] for utilization in the Rhie and Chow pressure velocity coupling as follows

$$u_f = u_f - \frac{1}{2} \left(\frac{\Omega_P F_P^x}{\alpha_P} + \frac{\Omega_E F_E^x}{\alpha_E} \right) + \frac{\Omega_f F_f^x}{\alpha_f}, \quad (4.35)$$

where $\alpha_i = \left[1 + \frac{\rho_i \Omega_i}{a_i \Delta t} \right]$.

4.4.2. COEFFICIENT CORRECTION

Van Wachem and Denner[12] proposed modifications to the pressure velocity coupling specific to two-phase flows to ensure that the discretization does not lead to any unphysical numerical disturbances.

MODIFICATIONS TO PRESSURE

The density at the face was calculated by a harmonic average of the density at the adjacent cell centers

$$\rho_f = \frac{2}{\rho_P^{-1} + \rho_{NB}^{-1}}, \quad (4.36)$$

where NB is the neighboring cell.

Furthermore, the pressure terms corresponding to the cell centers were weighted by the ratio of the density at the face and the density at the cell center. Hence equation 4.7 simplifies to

$$u_f = \frac{u_P + u_E}{2} - \left[\frac{1}{2} \left(\frac{\Omega_P}{a_P} + \frac{\Omega_E}{a_E} \right) p_{,1} \right]_f - \rho_f \left(\frac{\Omega_P}{a_P \rho_P} p_{,1} \Big|_P + \frac{\Omega_E}{a_E \rho_E} p_{,1} \Big|_E \right) - \left[\left(\frac{\rho_P \Omega_P}{a_P \Delta t} u_P^n + \frac{\rho_E \Omega_E}{a_E \Delta t} u_E^n \right) - \frac{1}{2} \left(\frac{\rho_P \Omega_P}{a_P \Delta t} + \frac{\rho_E \Omega_E}{a_E \Delta t} \right) u_f^n \right]. \quad (4.37)$$

MODIFICATIONS TO GRAVITY

The source term attributed to the gravity is similarly weighted by the ratio of the density at the face to the density at the cell center for a consistent discretization. Furthermore, discretely gravity has to be implemented on the same stencil as the pressure gradient to ensure the discrete force balance.

The gravity source term is written as

$$u_f = u_f - \frac{\rho_f}{2} \left(\frac{\Omega_P F_P^y}{\alpha_P \rho_P} + \frac{\Omega_E F_E^y}{\alpha_E \rho_E} \right) + \frac{\Omega_f F_f^y}{\alpha_f}, \quad (4.38)$$

where

$$F_p = \sum_P (g \cdot a_f) \rho_f n_f A_f, \quad (4.39)$$

$$a_f = \mathbf{X}_f - \mathbf{X}_{NB} \quad (4.40)$$

and A_f is the area of the face.

4.5. OVERVIEW

The Rhie and Chow pressure velocity coupling is widely used to strongly couple the pressure and the velocity field for a collocated arrangement of unknowns. The assumptions made in the method have no physical basis and may not hold true for all flow problems. Hence a careful formulation of the Rhie and Chow approach must be employed to ensure a consistent and well posed numerical implementation.

5

DISCRETE FORCE BALANCE

As discussed in Chapter 2, the mass flow rate and the momentum fields in multiphase flows have C^{-1} continuity at the interface due to the fluid property variation in the computational domain. Violation of the continuity and smoothness requirements of the discretization can lead to a discrete force imbalance producing *numerical* waves at the interface. Hence to accurately model flows at the onset of instability a special discretization needs to be utilized at the interface such that the discretization complies with the discontinuities, which are *a priori* known from the interface jump conditions. This chapter discusses the different strategies of discretization to achieve a discrete force balance.

5.1. GRAVITY CONSISTENT DISCRETIZATION

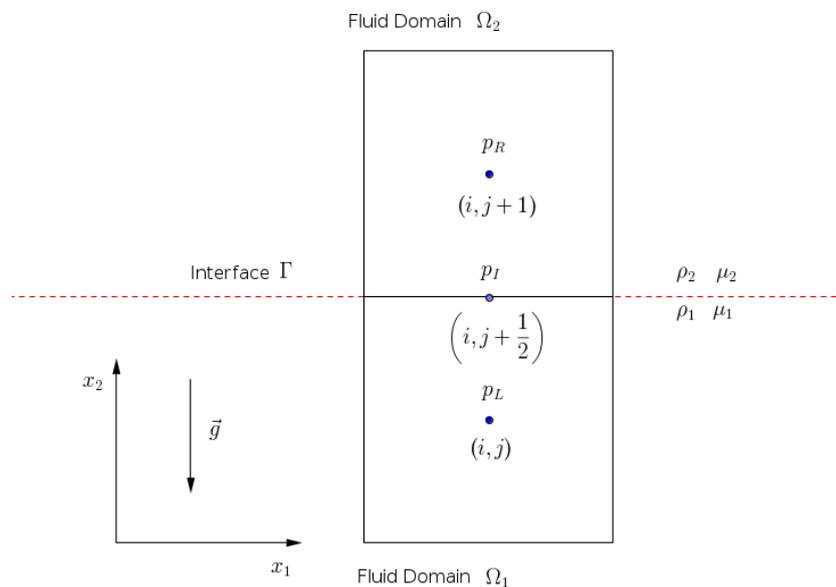


Figure 5.1: Example of a 2D computational grid with collocate arrangement of unknowns, with coincident interface and cell edge

In quiescent two phase flow on a computational domain as shown in figure 5.1, the hydrostatic pressure in the fluid is balanced by the weight of the fluid, which is a result of the gravitational acceleration. Hence the pressure field complies with

$$\frac{\nabla p}{\rho} = \mathbf{g}. \quad (5.1)$$

As a result the pressure in the fluid is given by

$$p_1(x_1, x_2) = \rho_1 \mathbf{g} x_2, \quad \forall x_2 \in \Omega_1, \quad (5.2)$$

$$p_2(x_1, x_2) = \rho_2 \mathbf{g} x_2, \quad \forall x_2 \in \Omega_2. \quad (5.3)$$

Moreover, interface jump conditions dictate the pressure at the interface in both the fluids will be equal, hence

$$p_1^\Gamma(x_1, x_2) = p_2^\Gamma(x_1, x_2). \quad (5.4)$$

Equation (5.1) implies

$$\frac{\partial p_1^\Gamma}{\partial x_2} \neq \frac{\partial p_2^\Gamma}{\partial x_2}. \quad (5.5)$$

A modified pressure can be defined such that its discretization satisfies properties (5.4) and (5.5). A discretization that satisfies these properties is commonly referred to as the Gravity Consistent Pressure Discretization [17].

5.1.1. DISCRETE MOMENTUM EQUATION

Consider a 2D computational domain as shown in Figure 5.1. The interface coincides with the north cell face having face center $\mathbf{x}(i, j + \frac{1}{2})$ of the cell with cell center $\mathbf{x}(i, j)$. Discretizing the x_2 momentum equation spatially on cell (i, j) we get

$$(\rho u_2 u_2 - \mu u_{2,1}) \Big|_{i-\frac{1}{2}, j}^{i+1, j} + (\rho u_1 u_2 - p - \mu u_{2,2}) \Big|_{i, j-\frac{1}{2}}^{i, j+\frac{1}{2}} = 0. \quad (5.6)$$

The discretization requires values of pressure to be evaluated at the face centers. As the pressure unknowns are located at the cell centers, it is linearly interpolated to evaluate the value at the face as

$$p_{i, j+\frac{1}{2}} = \frac{p_{i, j} + p_{i, j+1}}{2}. \quad (5.7)$$

As discussed previously, the pressure varies *piecewise* linearly in a quiescent flow. Hence linear interpolation is not valid and may produce a numerical force unbalance. Instead a Gravity Consistent Pressure discretization can be utilized as follows:

- **Modified pressure**

A modified pressure [18] is defined as the summation of the pressure and the gravity term. The open-source OpenFOAM solver *interFoam* utilizes this method [19]. A similar definition was employed by Rusche in his work on bubbly flows [18].

- **Ghost Fluid Method**

Fedkiw et al. proposed a Ghost Fluid Method [20], [21] for two phase flow with a material interface represented by the Level Set function. GFM defines a ghost fluid having properties of fluid 2 where fluid 1 is present and vice versa. The conservation equations are solved for both the real and the ghost fluid. Then the system is advanced in time and the appropriate choice between the two is made according to the sign of the signed distance function. Kang et al. [3] devised a GFM based Boundary Condition Capturing method, which implements a sharp representation of the interface using the interface jump conditions. The method interpolates an augmented flow field across the interface to account for the discontinuity.

- **Face value reconstruction based on second order Taylor expansion**

Queutey, Visonneau reconstructed face values of flow variables on a collocated grid employing a second order Taylor expansion based on *a priori* known flow discontinuities assuming the material interface coincides with the faces of the control volumes in the mesh [1].

5.2. MODIFIED PRESSURE

A modified pressure is defined as

$$p^m = p - \rho \mathbf{g} \cdot \mathbf{X}. \quad (5.8)$$

Hence

$$\nabla p^m = \nabla p - \rho \mathbf{g} - \mathbf{g} \cdot \mathbf{X} \nabla \rho \quad (5.9)$$

$$\nabla p = \nabla p^m + \rho \mathbf{g} + \mathbf{g} \cdot \mathbf{X} \nabla \rho. \quad (5.10)$$

Utilizing the modified pressure, the momentum equation is modified as

$$\frac{\partial \rho \mathbf{U}}{\partial t} + \nabla \cdot (\rho \mathbf{U} \mathbf{U}) = -\nabla p^m - (\mathbf{g} \cdot \mathbf{X}) \nabla \rho + \nabla (\mu (\nabla \mathbf{U} + \nabla \mathbf{U}^T)). \quad (5.11)$$

5.2.1. MOTIVATION

Nodal pressure values are defined on a collocated grid, hence the pressure has to be approximated at the face centers. The modified pressure is obtained by subtracting the gravity term from the pressure, which 'removes' the discontinuous step gravity term (making the modified pressure continuous instead of piecewise continuous for quiescent case), which allows a simple averaging at the face center.

5.2.2. MATHEMATICAL VALIDITY OF MODIFIED MOMENTUM EQUATION

In most multiphase flow numerical frameworks the density is regularized producing a smooth density field as a function of the signed distance function in the Level Set method or as a function of the volume fraction in the Volume of Fluid Method. Such a diffused representation of the interface produces a smeared density profile at the interface, mathematically permitting the definition of the density gradient at the interface, which otherwise does not exist. Conversely a sharp representation maintains a sharp distinction between two fluids. As a result, at the interface the gradient of density does not exist, and hence the modified pressure cannot be used for a sharp representation of the interface.

5.3. GHOST FLUID METHOD

Fedkiw et al. [20] devised a numerical method called the Ghost Fluid Method (GFM) for multiphase flows with an interface, formulated such that a multiphase flow problem can be solved with a single phase fluid solver. It employs two sets of flow values, one set having mass and momentum of the real fluid at that point, and another set of a ghost mass and momentum having properties of the fluid across the interface. Both sets of equations are advanced in time, and the appropriate choice is made between the two according to the sign of the signed distance to the interface at the new time level. The ghost values of a continuous function are defined to be equal to the value of the real fluid at that point, and the value for a discontinuous function is extrapolated to the ghost cell using a one sided approximation. The motivation for this is based on an implicit accountability of the boundary conditions at the interface for contact discontinuities.

Hence, for a discontinuous quantity Q , first the normal to the interface is calculated using the level set function

$$\mathbf{N} = \frac{\nabla \phi}{|\nabla \phi|}. \quad (5.12)$$

Then the quantity is extrapolated as

$$Q_t \pm \mathbf{N} \cdot \mathbf{Q} = 0. \quad (5.13)$$

As an example consider the stress tensor in a viscous flow. The GFM splits it in a continuous and a discontinuous part. The ghost value of the stress tensor is obtained by adding the continuous part of stress tensor of the real fluid and the discontinuous part of the extrapolated ghost value. The interested reader is referred to [21] for more details.

5.3.1. BOUNDARY CONDITION CAPTURING GHOST FLUID METHOD

Kang et al. proposed a finite difference method with a sharp treatment for the interface on a staggered grid that employs GFM called the Boundary Condition Capturing Ghost Fluid method (BCC-GFM) [20]. BCC-GFM explicitly solves for flow variables at the interface utilizing the interface jump conditions.

As an example consider the viscous terms (5.14) and (5.15) in the momentum equations. Components of the first derivatives of the velocity are discontinuous across the interface. Therefore the interface jump conditions are employed for the discretization at the interface [3] to account for the discontinuity:

$$\frac{\mu}{\rho}(u_{1,\alpha\alpha}) \quad (5.14)$$

$$\frac{\mu}{\rho}(u_{2,\alpha\alpha}) \quad (5.15)$$

Equations (2.27), (2.28), (2.29), (2.30) are used to derive the following jump condition

$$\begin{pmatrix} [\mu u_{1,\alpha}]^T \\ [\mu u_{2,\alpha}]^T \\ [\mu u_{3,\alpha}]^T \end{pmatrix} = [\mu] \begin{pmatrix} \nabla u_1 \\ \nabla u_2 \\ \nabla u_3 \end{pmatrix} \begin{pmatrix} \mathbf{0} \\ \mathbf{T}_1 \\ \mathbf{T}_2 \end{pmatrix} \begin{pmatrix} \mathbf{0} \\ \mathbf{T}_1 \\ \mathbf{T}_2 \end{pmatrix}^T + [\mu] \mathbf{NN}^T \begin{pmatrix} \nabla u_1 \\ \nabla u_2 \\ \nabla u_3 \end{pmatrix} \mathbf{NN}^T - [\mu] \begin{pmatrix} \mathbf{0} \\ \mathbf{T}_1 \\ \mathbf{T}_2 \end{pmatrix} \begin{pmatrix} \mathbf{0} \\ \mathbf{T}_1 \\ \mathbf{T}_2 \end{pmatrix}^T \begin{pmatrix} \nabla u \\ \nabla u_2 \\ \nabla u_3 \end{pmatrix}^T \mathbf{NN}^T. \quad (5.16)$$

Note that the derivatives of the components of the velocity field in equation (5.16) have been expressed as functions of the continuous fields only.

Equation (5.16) is rewritten as shown in equation 5.17 for a easier reference needed at a later stage

$$\begin{pmatrix} [\mu u_{1,\alpha}]^T \\ [\mu u_{2,\alpha}]^T \\ [\mu u_{3,\alpha}]^T \end{pmatrix} = \begin{bmatrix} j^{11} & j^{12} & j^{13} \\ j^{21} & j^{22} & j^{23} \\ j^{31} & j^{32} & j^{33} \end{bmatrix}. \quad (5.17)$$

DISCRETIZATION OF VISCOUS TERM

The discretization of the viscous term $\mu u_{1,11}$ will be discussed to demonstrate the details of the Boundary Condition Capturing method. Consider a cell having center (i, j) intersected by an interface, the level set function (ϕ) has opposite signs at the two vertical edges, i.e.

$$\phi_l = \phi_{i-\frac{1}{2},j} < 0,$$

$$\phi_r = \phi_{i+\frac{1}{2},j} > 0.$$

The interface splits the cell in $\theta\Delta X$ and $(1-\theta)\Delta X$, such that

$$\theta = \frac{|\phi_l|}{|\phi_l| + |\phi_r|}. \quad (5.18)$$

The velocity is continuous but the first derivatives are discontinuous across the interface, hence $(\mu u_{1,1})_l$ is explicitly calculated using interface jump condition (5.16) and (5.17). The jump across the interface is calculated as

$$J_I = \theta J_r + (1-\theta)J_l. \quad (5.19)$$

The jump across the interface $[\mu u_{1,1}] = J_I$ is expressed in terms of the velocities at the cell centers and the interface as

$$J_I = \mu^+ \left(\frac{u_r - u_I}{(1-\theta)\Delta x} \right) - \mu^- \left(\frac{u_I - u_l}{\theta\Delta x} \right). \quad (5.20)$$

Here μ^\pm are the viscosities in the respective domains and the velocities $u_r = u_{i+\frac{1}{2},j}$, $u_l = u_{i-\frac{1}{2},j}$.

Next, equation (5.20) is solved for u_I which is the interface velocity. The calculated interface velocity is then used to calculate $(\mu u_x)_l$ as

$$(\mu u_{1,1})_l = \mu^+ \left(\frac{u_r - u_I}{(1-\theta)\Delta x} \right) = \hat{\mu} \left(\frac{u_r - u_l}{\Delta x} \right) - \frac{\hat{\mu} J_I \theta}{\mu^+}, \quad (5.21)$$

where,

$$\hat{\mu} = \frac{\mu^- \mu^+}{\mu^- \theta + \mu^+ (1 - \theta)}. \quad (5.22)$$

For more details the reader is referred to [3].

5.4. FACE VALUE RECONSTRUCTION BASED ON SECOND ORDER TAYLOR EXPANSION

Queutey and Visonneau proposed and implemented an interface capturing method (ICM) based on a face reconstruction of the flow values using a second order Taylor expansion under the assumption that the interface coincides with the faces of the grid cells in the mesh [1].

For a flow quantity Q the reconstruction is carried out based on the *a priori* known interface jump for a quantity and its gradient

$$[Q] = a, \quad (5.23)$$

$$[c\nabla Q] = \mathbf{b}. \quad (5.24)$$

Assuming that the interface coincides with the faces of the cells in the computational domain, the jump is approximated as

$$[Q] = Q^+ - Q^- \simeq Q_{f^+} - Q_{f^-}. \quad (5.25)$$

where

$$Q_{f^+} = \lim_{\mathbf{X} \rightarrow \Gamma} Q^+(\mathbf{X}), \quad (5.26)$$

$$Q_{f^-} = \lim_{\mathbf{X} \rightarrow \Gamma} Q^-(\mathbf{X}). \quad (5.27)$$

Refer Figure 5.3 for the definition of the vectors.

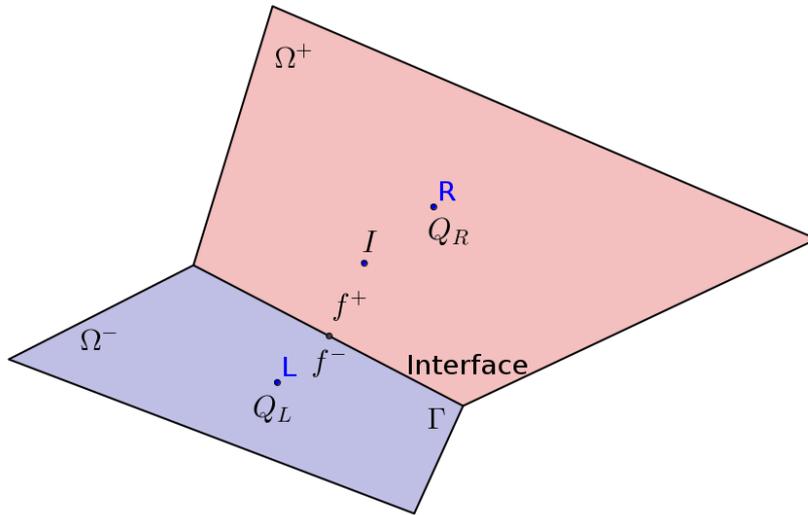


Figure 5.2: Computational grid with adjacent cells L , R with coincident interface on face f

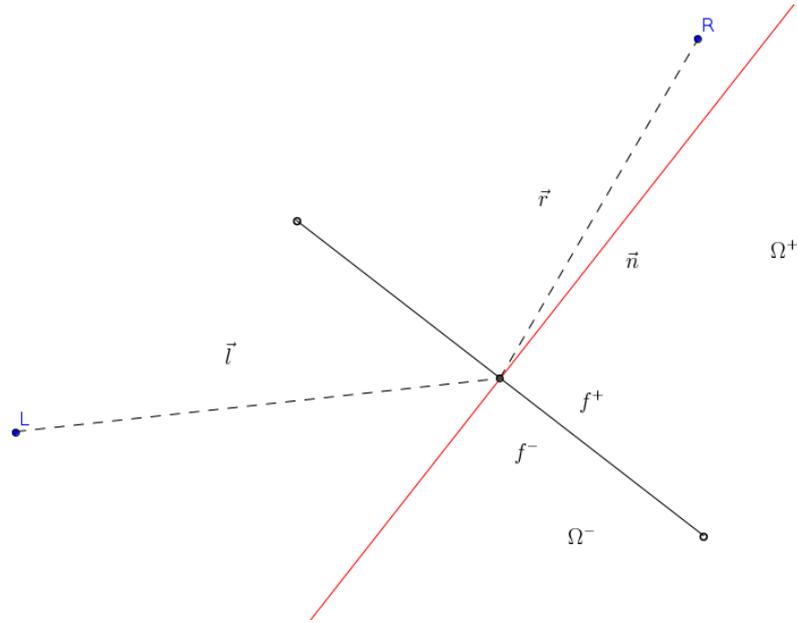


Figure 5.3: Computational grid with adjacent cells L, R with coinciding interface with the face f

Retaining a second order Taylor expansion at L and R

$$Q_L \approx Q_{f^-} - \mathbf{l} \cdot \nabla Q_{f^-}, \quad (5.28)$$

$$Q_R \approx Q_{f^+} + \mathbf{r} \cdot \nabla Q_{f^+}. \quad (5.29)$$

Equations (5.28), (5.29) and (5.23), (5.24) are utilized to solve for the face values Q_{f^+} , Q_{f^-} . Refer to [1] for more details.

As an example the pressure in a quiescent flow is subject to:

$$[p] = 0, \quad (5.30)$$

$$\left[\frac{\nabla p \cdot \mathbf{n}}{\rho} \right] = 0. \quad (5.31)$$

ICM uses (5.30) and (5.31) for calculating the value of the pressure at the interface.

ICM requires a high density of grid cells near the interface to accurately capture the interface, as the solver assumes that the interface coincides with the mesh. The strict requirement of the interface coinciding limits the applicability of the method and necessitates the use of computationally expensive Adaptive Mesh refinement techniques to reduce the impact of the assumption.

5.5. COMPARISON BETWEEN BALANCED-FORCE METHODS

The table below compares the discussed three methods to achieve a discrete force balance.

Method	Comments
Modified Pressure	<ul style="list-style-type: none"> - A modified pressure is defined which includes the gravity term in the formulation. -Modified pressure adds an extra term of the density gradient to the momentum equation. The normal component of the density gradient does not exist for a two-phase flow at the interface, due to the discontinuity in the density.
Boundary Capturing Ghost Fluid Method (Kang et al)	<ul style="list-style-type: none"> -Discontinuous flow fields are augmented by explicitly incorporating the interface jump conditions. -BCC-GFM was formulated for a discretization using a staggered arrangement of the unknowns.
Interface Capturing Method (Queutey, Vissonneau)	<ul style="list-style-type: none"> - Field values needed at a cell face for flux calculations are found employing a second order Taylor Expansion based on interface jump conditions. - The formulation is based on the assumption that the interface coincides with the faces of the control volumes in the computational domain, limiting the applicability of the method.

6

NUMERICAL CHALLENGES IN MODELING FLOW AT THE ONSET OF INSTABILITY

In a finite volume method the flow domain is decomposed into a finite number of control volumes, on which the unknowns are arranged in a collocated or a staggered way. The governing flow conservation equations are solved to obtain the solution with a certain order of accuracy. Multiphase flows can have a wide range of spatial and temporal characteristics according to the multiphase flow regime and the multiphase flow application. Hence numerically modeling multiphase flows requires special treatment specific to the dominating dynamics in the flow. The focus of this research is the numerical modeling of incompressible, immiscible two phase-flows at the onset of instability. To accurately predict the flow numerically the following properties have to be satisfied:

- **Discrete conservation of mass and momentum**
- **Discrete force balance**

Numerical challenges that are faced in an attempt to satisfy the above properties were discussed in Chapter 1. Chapters 1-4 discussed and analyzed the building blocks of a numerical framework for simulating multiphase flows. In this Chapter, the bottlenecks in the numerical framework are summarized and reviewed.

The numerical challenges can be categorized according to the form of the governing equations: the continuous form and the discrete form of the governing equations.

6.1. CONTINUOUS FORM OF GOVERNING EQUATIONS

Conservation of mass and momentum in the continuous form are the basis of numerically solving a flow problem. The continuous form of these equations was analyzed in Chapter 2, which will be reviewed below.

- **Form of governing equations**

The momentum equation can be formulated in a conservative or a non conservative form. Further, the pressure term can be incorporated in the gradient viscous term as described in section 2.5. The terms involved in the formulation may represent physical quantities such as mass, momentum and shear stress or purely mathematical variables. Discretization of the governing equation conserves mass and momentum numerically and the effect on the conservation according to the nature of the quantities is unclear. Therefore, careful discretization needs to be employed to ensure that the choice of the form of continuous equation does not lead to numerical errors.

- **Discontinuity in flow fields**

In multiphase flows a jump in the density creates a discontinuity in the momentum field as discussed in 2.3 and as a result the momentum equation does not hold at the interface. Hence the governing equations have to be solved away from the interface coupled by the interface jump conditions.

6.2. DISCRETE FORM OF GOVERNING EQUATIONS

6.2.1. DISCONTINUITIES IN FLOW FIELDS

Multiphase flows have C^{-1} continuity due to density jump across the interface. Most methods regularize the density field producing a smooth density and flow fields permitting straightforward discretization, but these approximations lead to inaccurate numerical solutions for applications such as the onset of instabilities and lateral roll of a ship.

1. Interface Jump conditions

Interface jump conditions describe the physics of the flow and the nature of discontinuities at the interface. A straightforward discretization disregards these jump conditions, which creates an unbalance in the forces acting at the interface leading to a physically unrealistic numerical solution.

2. Sharp representation of the interface

Most of the numerical methods regularize discontinuous flow fields and smear out the interface over a few cells. Such an approach is satisfactory for flows having a relatively large velocity and length scales, regularizing the density field smoothens the interface over few a control volumes, which is small in comparison with the length scale of the flow structures. On the other hand flow at the onset of instability at the interface is initiated with very small physical perturbations and hence requires an accurate consistent numerical treatment to avoid adding numerical artefacts.

A proper discretization should respect the distinction between the two fluids and should maintain a sharp interface to produce a physically accurate solution.

6.2.2. CONSERVATION OF MASS AND MOMENTUM

Pressure velocity coupling

A collocated arrangement employing cell centered pressure and velocity values leads to a weakly coupled pressure and velocity field. Pressure-velocity coupling is employed to calculate the face velocity to impose conservation of mass. The Rhie and Chow pressure-velocity coupling constructs a face velocity under a set of assumptions described in 4.1. A careful formulation is needed for multiphase flows due to the presence of discontinuities at the interface.

6.2.3. DISCRETE FORCE BALANCE

Queuetey et al. [1] achieved a force balance by reconstruction of the face values based on a second order Taylor expansion. The method assumes that the interface coincides with the faces of the control volumes in the computational domain, limiting its applicability. The objective of the present research is to achieve a balanced-force framework for a general setting such that the formulation is applicable independent of the interface location and orientation.

7

RESEARCH PROPOSAL

7.1. RESEARCH OBJECTIVE

The subject of this research is the development of a balanced-force numerical framework for two-phase immiscible, incompressible flow to accurately model flow at the onset of instability. Gravity will be incorporated as a body force. Surface tension which acts as a surface force at the interface produces a discrete jump in the pressure across the interface and hence can be added *ad hoc* to the discretization; it does not necessitate a special treatment and therefore it will not be considered here. To enable an accurate prediction of the flow at the onset of instability the formulation should satisfy the following properties:

1. Produce a discrete force balance.
2. Conserve mass and momentum.
3. Applicable for an arbitrary interface, i.e. independent of position and orientation of the interface with respect to the computational grid.
4. Satisfy interface jump conditions.

A literature study of previous research work, described in Chapters 2-7, revealed the limitations of the current numerical methods. The ICM proposed by Queutey et al. [1] achieves an exact force balance when the interface coincides with the mesh. The method constructs the face values in a cell using a second order Taylor expansion from the known interface jump conditions. The strong assumption that the interface coincides with the cell faces limits the applicability and the accuracy of the method. The objective of this thesis is to propose a numerical framework that complies with the interface jump conditions applicable to arbitrarily located and oriented interface.

Similarly, Kang et al. [3] implemented the interface jump conditions using a finite difference method on a uniform staggered Cartesian grid by augmenting the discontinuous flow fields. The method accomplishes a discrete force balance, but a Cartesian staggered grid limits the applicability of the method to simple geometries. Hence in this thesis the numerical method will be proposed that does not exploit properties of a Cartesian grid and in principle is applicable for a general, non-uniform, unstructured collocated grid.

Most methods use a straightforward discretization disregarding interface jump conditions and regularize discontinuous flow fields. As a result of these approximations the flow solution does not converge to machine precision. The research question can then be posed, whether the numerical framework converges upto the machine precision if the flow field discontinuities are satisfied?

7.2. RESEARCH OUTLINE

The research will consist of proposing a numerical framework and testing the efficacy of the proposed method with a set of test cases.

The research will consist of the following stages:

1. Formulate a balanced-force numerical framework for calculating flow fields near the interface for a unstructured collocated grid.

2. Implement the numerical method for a 2D Cartesian collocated grid.
3. Test the numerical method for simple test cases

The complete framework will consist of different components that have a different function, such as pressure velocity coupling, calculation of values at the interface and at the face. These components will be tested individually: step by step by implementing the framework for specific flow problems to assess the effectiveness of each component. The candidate test cases are described below.

- (a) Test cases when the interface location is known

Interface methods employed to track the interface such as the Level Set Method and the Volume of Fluid Method have their complexities and inaccuracies. Hence the first step is to test the effectiveness of the method, without introducing the additional inaccuracies of the interface methods, i.e. it is assumed that the interface location is known.

- i. Quiescent Flow- Interface parallel to the mesh face

In a quiescent flow, both the fluids are stationary and the gravity is balanced by the pressure gradient. When the interface is aligned to the faces of the control volumes in the computational domain the face value reconstruction needs to be carried out in one Cartesian direction (normal to the interface) only. The test case will be subdivided into further configurations as:

- A. Test Case 1- Coincident interface with the mesh

When the interface coincides with the cell faces the face values are equal to the values at the interface which are calculated based on the interface jump conditions. This test case enables verification of the framework without involving the numerics to reconstruct the face values based on values at the interface. Figure 7.1 shows an example of a case where the interface is coinciding with the mesh.

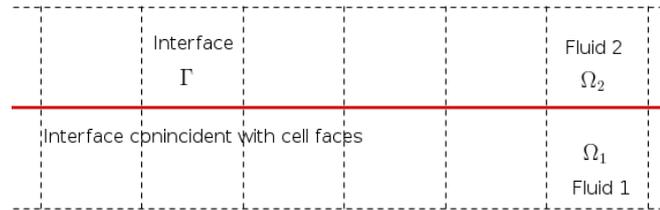


Figure 7.1: Example of a coinciding interface with the mesh

- B. Test Case 2- Interface offset to the cell face

When the interface is aligned but offset to the faces of the control volumes, first the interface values are calculated using the interface jump conditions, based on which the face values are reconstructed. This case will test the accuracy of the face reconstruction employed in the numerical framework. Figure 7.2 shows an example of such a case where the interface is parallel and offset to a row of grid cells.

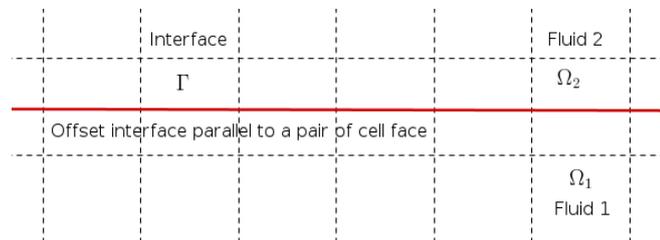


Figure 7.2: Example of an offset interface to the mesh

- ii. Quiescent Flow- Interface oriented to the cell

- A. Test Case 3-Interface cuts opposite side of the cell face

When the interface is oriented at a small angle such that the interface intersects cells in one row of cells and all the face centers on the vertical edges lie in one fluid only the calculation simplifies as face value reconstruction needs to be done in \hat{e}_2 coordinate direction only. Figure 7.3 shows such an example.

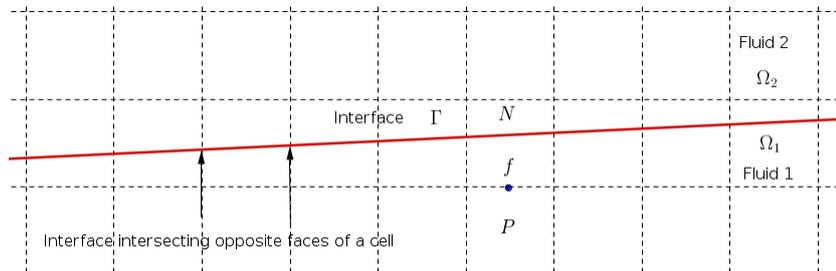


Figure 7.3: Example where the interface cuts two opposite faces of a cell

B. Test Case 4- Interface cuts two adjacent faces

When the interface is steeper, it will intersect cells in more than one row, and will have cells, in which the interface intersects two adjacent cell faces. Figure 7.4 shows such an example. In this case the cell faces values will be calculated based on the interface jump conditions in both the coordinate directions.

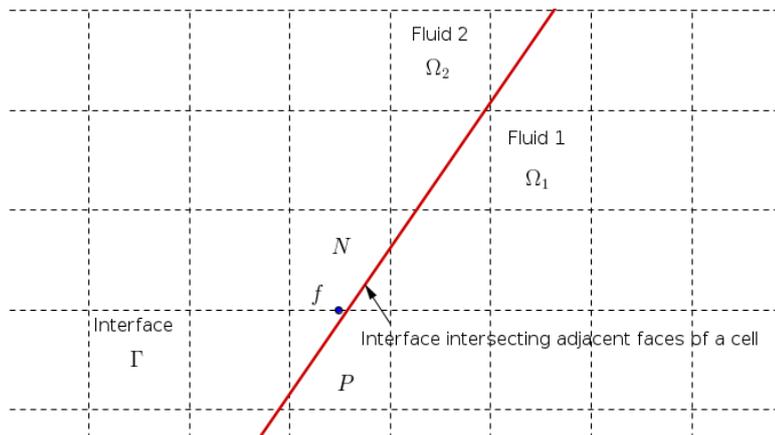


Figure 7.4: Example where the interface cuts two adjacent faces of the cell

(b) Test Cases where the interface location is not fixed

i. Test Case 5- Viscous Standing Wave

When the interface between the two fluids is perturbed, the interface undergoes oscillations before the perturbation damps out to reach the steady state of quiescent flow. A different oscillatory damping is observed when the Reynolds number of the flow is varied.

ii. Test Case 6- Flow over an obstacle

Flow over an obstruction creates waves at the interface. To accurately predict such a flow created the waves should be purely physical and not numerical. Hence accurately modeling flow at the onset of instability requires an accurate discrete force balance.

8

BALANCED-FORCE DISCRETIZATION: COINCIDING INTERFACE AND CELL FACE

The Interface Capturing method (ICM) for the collocated arrangement of unknowns as described in [1], which is applicable when the interface coincides with the faces of the control volumes in the computational domain, involves a reconstruction of values at the interface employing a second order Taylor expansion based on the *a priori* known magnitudes of the jumps in the values at the discontinuities. Furthermore, [1] does not provide the exact formulation of the operators G , B , DG^f and the diffusion operator J . The purpose of this chapter is to formulate these operators, such that they comply with the interface jump conditions by incorporating the face reconstruction for the case in which the interface coincides with the faces of the control volumes to produce a balanced-force formulation.

8.1. INTERFACE JUMP CONDITIONS

The complete dynamic interface jump condition for the pressure is given by

$$\left[\frac{p, \alpha}{\rho} \right] = \left[\frac{(2\mu u_\alpha)_{, \alpha} + (\mu(u_{\alpha, \beta} + u_{\beta, \alpha}))_{, \beta}}{\rho} \right]. \quad (8.1)$$

It is challenging to derive interface jump conditions for the second derivative of the velocity. The right hand side of (8.1) vanishes in the limit of the Reynolds number going to infinity. As a result the interface condition simplifies to the hydrostatic interface jump condition

$$\left[\frac{\nabla p \cdot \mathbf{g}}{\rho} \right] = 0. \quad (8.2)$$

For multiphase flows at high Reynolds number, the hypothesis is that the reconstruction of the pressure based on the hydrostatic assumption is sufficient to numerically model the dynamics at the interface. This is attributed to the fact that the variation in the velocity in the direction orthogonal to the interface is small, leading to small viscous stresses and pressure variations normal to the interface. Hence the change in pressure normal to the interface and the change in pressure in the direction of gravity will be dominated by the change in the hydrostatic pressure.

8.1.1. CANDIDATE TEST CASE

In a quiescent flow the system of equations simplifies as the velocity is zero and therefore the inertia term vanishes. Furthermore, the pressure gradient is balanced by gravity. The discretized terms in the operators G^c and G^f should cancel with gravity, and the operators DG^f and B should comply with the hydrostatic interface conditions to produce an exact force balance. If such a balance is achieved, the discrete system will satisfy the exact analytical solution, and if no significant numerical approximations or regularizations are made, the occurrence of numerical spurious velocities is avoided. A verification of an exact force balance for the operators DG^f and B is a necessary condition to display the effectiveness of the formulation for the reconstruction

based on the hydrostatic interface jump conditions. First, the formulation of the discrete operators for a uniform grid is presented, followed by results and discussion of the implementation for quiescent multiphase flow. Next the construction of the discrete operators on a non-uniform grid is presented, and the results of that implementation are discussed.

8.2. QUIESCENT FLOW: INTERFACE COINCIDING WITH CELL FACES ON A UNIFORM GRID

8.2.1. PRESSURE RECONSTRUCTION AT THE INTERFACE

In a quiescent flow the pressure is determined by gravity, and the velocities are zero. The pressure conforms to a linear distribution with a slope discontinuity at the interface. In this Section, the face reconstruction, based on the the ICM explained in Section 5, for a hydrostatic discontinuity in multiphase flows is discussed.

The interface jump conditions at the interface Γ for pressure are given by:

$$[p] = 0, \quad (8.3)$$

$$\left[\frac{\nabla p \cdot \hat{e}_2}{\rho} \right] = 0. \quad (8.4)$$

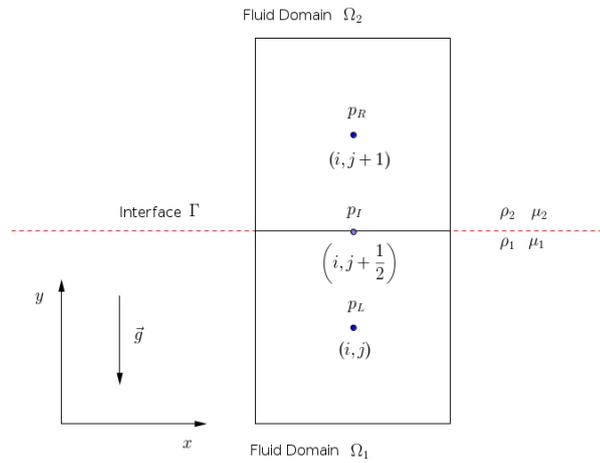


Figure 8.1: 2D Computational grid with the collocated arrangement of unknowns, in the case of coincident interface and cell edge

Pressure p_I at the interface is calculated based on the approach described in 5.4 as

$$p_L = p_I^- - \frac{h}{2} (\nabla^- p \cdot \hat{e}_2), \quad (8.5)$$

$$p_R = p_I^+ + \frac{h}{2} (\nabla^+ p \cdot \hat{e}_2). \quad (8.6)$$

Multiplying (8.6) by $\frac{1}{\rho_2}$ and (8.5) by $\frac{1}{\rho_1}$ and adding these two gives

$$\frac{p_L}{\rho_1} + \frac{p_R}{\rho_2} = \frac{p_I^-}{\rho_1} + \frac{p_I^+}{\rho_2} + \frac{h}{2} \left(\frac{\nabla^+ p \cdot \hat{e}_2}{\rho_2} - \frac{\nabla^- p \cdot \hat{e}_2}{\rho_1} \right). \quad (8.7)$$

and from (8.4),

$$\frac{\nabla^+ p \cdot \hat{e}_2}{\rho_2} = \frac{\nabla^- p \cdot \hat{e}_2}{\rho_1}. \quad (8.8)$$

Hence (8.7) simplifies to

$$\frac{p_L}{\rho_1} + \frac{p_R}{\rho_2} = \frac{p_I^-}{\rho_1} + \frac{p_I^+}{\rho_2}. \quad (8.9)$$

Further, from (8.3)

$$p_I^+ = p_I^-. \quad (8.10)$$

Hence

$$p_I = p_R \left(\frac{\rho_1}{\rho_1 + \rho_2} \right) + p_L \left(\frac{\rho_2}{\rho_2 + \rho_1} \right). \quad (8.11)$$

The derived pressure at the interface is utilized in the gradient operator where one of the faces coincides with the interface.

ANALYTICAL SOLUTION FOR PRESSURE

The pressure can be analytically calculated from

$$\frac{1}{\rho(\mathbf{x})} \nabla p \cdot \hat{\mathbf{e}}_2 = \mathbf{g}. \quad (8.12)$$

OPERATORS

As explained in Section 4.1.1, the operator \dot{B} is defined as

$$(\dot{B}p)_{\alpha \mathbf{i}}^\pm = \left[\frac{1}{2} \left(d_\alpha^h p_{\mathbf{i}} + d_\alpha^h p_{\mathbf{i} \pm \hat{\mathbf{e}}_\alpha} \right) - d_\alpha^{\frac{h}{2}} p_{\mathbf{i} \pm \frac{1}{2} \hat{\mathbf{e}}_\alpha} \right]. \quad (8.13)$$

The \dot{B} operator as explained (4.19), can be split as

$$(\dot{B}p)_{\alpha \mathbf{i}} = \dot{B}_{\alpha \mathbf{i}}^c p + \dot{B}_{\alpha \mathbf{i}}^f p, \quad (8.14)$$

where,

$$\dot{B}_{\alpha \mathbf{i}}^c p = \frac{1}{2} \left(d_\alpha^h p_{\mathbf{i}} + d_\alpha^h p_{\mathbf{i} \pm \hat{\mathbf{e}}_\alpha} \right), \quad (8.15)$$

$$\dot{B}_{\alpha \mathbf{i}}^f p = -d_\alpha^{\frac{h}{2}} p_{\mathbf{i} \pm \frac{1}{2} \hat{\mathbf{e}}_\alpha} \quad (8.16)$$

Divergence of the local operator \dot{B} gives,

$$\begin{aligned} Bp &= D(\dot{B}p) = D(\dot{B}^c p + \dot{B}^f p) \\ &= DB^c p + D\dot{B}^f p = B^c p + DG^f p. \end{aligned} \quad (8.17)$$

EXPECTED SOLUTION FOR DISCRETE OPERATORS

The iterative approach of Miller and Schmidt was discussed in section 4.2. The discrete system is initialized according to (8.12). From Section 4.2, it can be seen that for a quiescent flow the face and the cell center velocities have to be zero, therefore the discrete operators have to satisfy

$$DG^f p = 0, \quad (8.18)$$

$$B^c p = 0. \quad (8.19)$$

8.2.2. BOUNDARY CONDITIONS

B^c and DG_f operators were implemented for the case shown in Figure 8.2. The interface coincides with the cell face at $x_2 = 0.5$. The imposed boundary conditions are shown in Figure 8.2. The discretization has to be modified for the control volumes labeled (and other equivalent control volumes) in the figure to account for the boundary conditions and/or the interface.

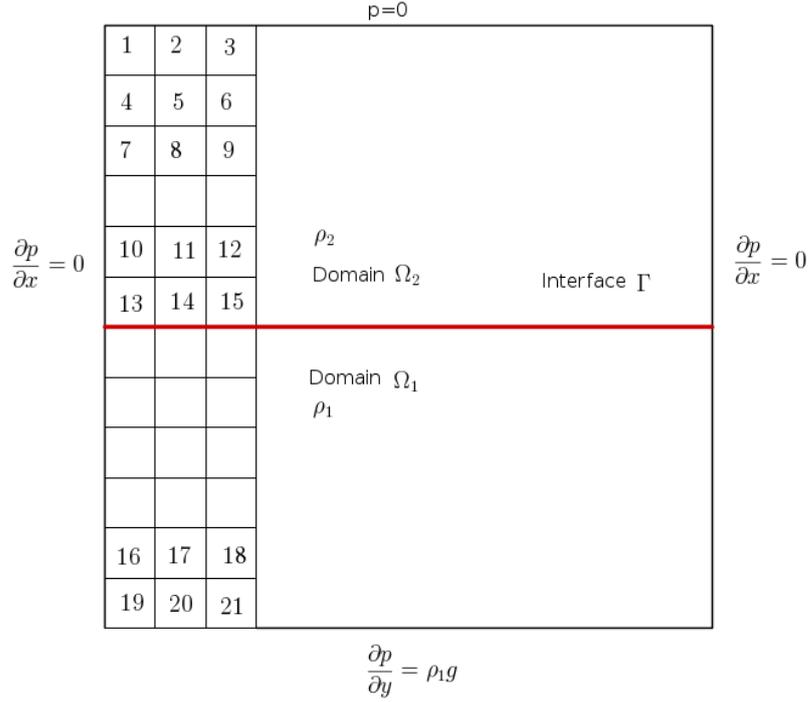


Figure 8.2: Implemented computational Grid with coincident interface and cell face

To achieve a discrete force balance, the operators involved in the pressure-velocity coupling should incorporate the value of the pressure at the interface derived through the reconstruction. Furthermore, the operators have to be formulated such that the operators comply with the interface jump conditions. In the following sections, a formulation based on these requirements is proposed, with the goal to obtain the discrete force balance.

8.2.3. OPERATOR B

The stencil of the operator B involves a 9 points stencil, hence the operator has to be modified for two rows of control volumes above and below the interface. Farther away from the interface, the stencil reduces to the single phase flow stencil. In the following subsections the discretization near and away from the interface is described.

FAR REMOVED FROM THE INTERFACE

Discretizing operator B gives

$$\begin{aligned}
 B^h p = h\Delta t & \left[\left(\frac{1}{2} (d_1^h p_i + d_1^h p_{i+\hat{e}_1}) - d_1^{\frac{h}{2}} p_{i+\frac{1}{2}\hat{e}_1} \right) - \left(\frac{1}{2} (d_1^h p_i + d_1^h p_{i-\hat{e}_1}) - d_1^{\frac{h}{2}} p_{i-\frac{1}{2}\hat{e}_1} \right) \right] \\
 & + h\Delta t \left[\left(\frac{1}{2} (d_2^h p_i + d_2^h p_{i+\hat{e}_2}) - d_2^{\frac{h}{2}} p_{i+\frac{1}{2}\hat{e}_2} \right) - \left(\frac{1}{2} (d_2^h p_i + d_2^h p_{i-\hat{e}_2}) - d_2^{\frac{h}{2}} p_{i-\frac{1}{2}\hat{e}_2} \right) \right].
 \end{aligned} \tag{8.20}$$

Away from the interface, the values of the pressure at the cell faces at $i \pm \frac{1}{2}$, $i \pm \frac{3}{2}$, $j \pm \frac{1}{2}$, $i \pm \frac{3}{2}$ are calculated by taking the average of the values at the adjacent cell centers. The discretization above gives the following 9 point stencil for a control volume away from the interface:

$$[B^h] = -\frac{\Delta t}{\rho} \begin{bmatrix} & & \frac{1}{4} & & \\ & & -1 & & \\ \frac{1}{4} & -1 & 3 & -1 & \frac{1}{4} \\ & & -1 & & \\ & & \frac{1}{4} & & \end{bmatrix}. \tag{8.21}$$

CELLS HAVING THE INTERFACE COINCIDING WITH A CELL FACE

Near the interface, the stencil is modified to account for the reconstruction at the interface, given by (8.11). Consider the pressure-correction applied to the control volume L with center (i, j) as shown in figure 8.1:

$$\dot{B}^{j+\frac{1}{2}} = \Delta t \left[\left(\frac{1}{2} \left(d_2^h p_{\mathbf{i}} + d_2^h p_{\mathbf{i}+\hat{\mathbf{e}}_2} \right) - d_1^{\frac{h}{2}} p_{\mathbf{i}+\frac{1}{2}\hat{\mathbf{e}}_2} \right) \right]. \quad (8.22)$$

The derivatives at the cell centers are evaluated by employing the pressure calculated from the pressure reconstruction:

$$\dot{B}_c^{j+\frac{1}{2}} p = \frac{1}{2} \left(d_2^h p_{\mathbf{i}} + d_2^h p_{\mathbf{i}+\hat{\mathbf{e}}_2} \right) \quad (8.23)$$

$$d_2^h p_{\mathbf{i}} = \frac{p_I - \frac{p_{i,j} + p_{i,j-1}}{2}}{\rho_1 h}, \quad (8.24)$$

$$d_2^h p_{\mathbf{i}+\hat{\mathbf{e}}_2} = \frac{\frac{p_{i,j+2} + p_{i,j+1}}{2} - p_I}{\rho_2 h}. \quad (8.25)$$

The formulation described above ensures a discrete force balance because the projection of the gradient of the pressure in direction $\hat{\mathbf{e}}_2$ scaled by the inverse of density, equals gravity. Near the interface this is achieved due to the fact that the reconstructed value of the pressure at the interface (hence, in this case the pressure on the face) is employed, i.e. the approximation of the pressure gradient over the slope discontinuity is avoided.

8.2.4. OPERATOR DG^f

The projections of the pressure gradients at the face are evaluated in the operator DG^f . Similar to the formulation of operator B , the discrete operator should be constructed such that it satisfies the interface jump conditions and incorporates the reconstructed pressure value. In the following sections, the formulation of the operator is explained for the case of control volumes near and far away from the interface.

FAR REMOVED FROM THE INTERFACE

From the definition 8.2.1, operator DG^f is given by

$$DG^f p = h\Delta t \left[\left(d_1^{\frac{h}{2}} p_{\mathbf{i}+\frac{1}{2}\hat{\mathbf{e}}_1} - d_1^{\frac{h}{2}} p_{\mathbf{i}-\frac{1}{2}\hat{\mathbf{e}}_1} \right) - \left(d_2^{\frac{h}{2}} p_{\mathbf{i}+\frac{1}{2}\hat{\mathbf{e}}_2} - d_2^{\frac{h}{2}} p_{\mathbf{i}+\frac{1}{2}\hat{\mathbf{e}}_2} \right) \right]. \quad (8.26)$$

Discretizing this gives

$$DG^f p = h\Delta t \left[\frac{p_{i+1,j} - p_{i,j}}{\rho h} + \frac{p_{i,j} - p_{i-1,j}}{\rho h} + \frac{p_{i,j+1} - p_{i,j}}{\rho h} - \frac{p_{i,j} - p_{i,j-1}}{\rho h} \right]. \quad (8.27)$$

$$= h\Delta t \left[\frac{p_{i+1,j} + p_{i,j+1} + p_{i-1,j} + p_{i,j-1} - 4p_{i,j}}{\rho h} \right] \quad (8.28)$$

The stencil for DG far removed from the interface is given by:

$$\left[DG^f h \right] = -\frac{\Delta t}{\rho} \begin{bmatrix} & -1 & \\ -1 & 4 & -1 \\ & -1 & \end{bmatrix}. \quad (8.29)$$

CELLS HAVING THE INTERFACE COINCIDING WITH A CELL FACE

Consider the discretization of the operator DG^f for control volume L in figure 8.1. To avoid approximating the pressure gradient over the discontinuity near the interface, a one-sided approximation is employed as follows

$$d_2^{\frac{h}{2}} p_{\mathbf{i}+\frac{1}{2}\hat{\mathbf{e}}_2} = \frac{p_I - p_{i,j}}{\rho_1 \frac{h}{2}}, \quad (8.30)$$

where p_I is the pressure at the interface calculated in equation 8.11. Here, it can be inferred that such a formulation ensures that the operator complies with the interface jump conditions.

8.2.5. RESULT

Operators DG^f and B^c were implemented and tested for varying density ratios and mesh sizes ($N = n \text{ times } n$). Table 8.1 shows the L^2 norm of the residuals for the implementation. The vectors r_{DG} and r_B arise out of the imposed inhomogeneous boundary conditions.

Table 8.1: L^2 norm of the residuals

Density Ratio	Mesh	$\frac{\ DG^f h p - r_{DG^f}\ _2}{N}$	$\frac{\ B^h p - r_B\ _2}{N}$
$\frac{\rho_2}{\rho_1} = 1$	n=8	0	0
	n=16	0	0
	n=32	0	0
$\frac{\rho_2}{\rho_1} = 100$	n=8	2.7 e-14	5.0 e-14
	n=16	1.1 e-14	1.7 e-14
	n=32	4.4 e-14	6.6 e-14
$\frac{\rho_2}{\rho_1} = 1000$	n=8	4.3 e-14	5.6 e-14
	n=16	1.2 e-14	2.1 e-14
	n=32	4.4 e-14	7.1 e-14
	n=64	1.7 e-14	1.7 e-14
	n=128	6.8 e-14	9.5 e-14

It can be seen that the norm of the residual is independent of the density ratio and the mesh size. The calculated residual is upto machine precision. Hence, it can be concluded that the proposed formulation of the operators satisfies the hydrostatic discontinuity and produces an exact force balance.

8.3. QUIESCENT FLOW: INTERFACE COINCIDING WITH THE FACES OF THE CONTROL VOLUMES ON A NON-UNIFORM GRID

The pressure-correction involves calculating the projected pressure gradients at the face and at the cell center of the control volume. If the flow domain is discretized on a unstructured or a non-uniform grid, the discrete operators are evaluated at different spatial scales, which may introduce errors in the numerically calculated pressure gradient at the interface. To investigate the effect of the discretization on an unstructured grid, the operators were implemented on a non-uniform grid as shown in figure 8.3. The mesh size of the control volumes normal to the interface is varied.

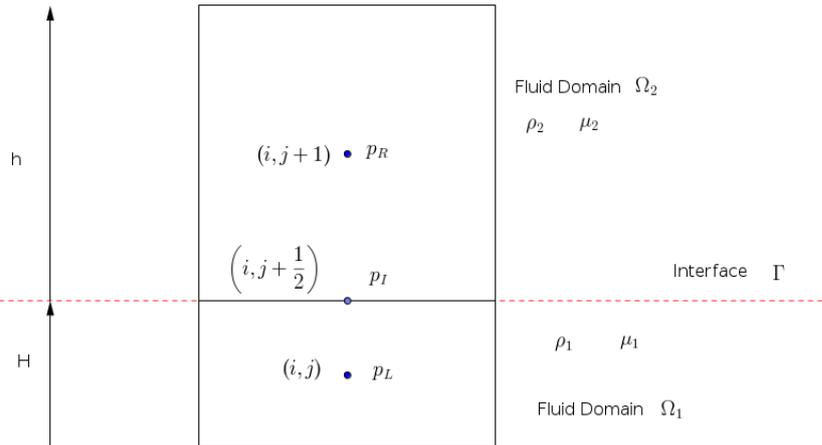


Figure 8.3: 2D Computational non-uniform grid with collocated arrangement of unknowns

8.3.1. PRESSURE RECONSTRUCTION AT THE INTERFACE

Employing the pressure reconstruction at the interface for a non-uniform grid gives

$$p_I = p_R \left(\frac{\rho_1 h}{\rho_1 h + \rho_2 H} \right) + p_L \left(\frac{\rho_2 H}{\rho_2 H + \rho_1 h} \right). \quad (8.31)$$

8.3.2. DISCRETE OPERATORS

An approach similar to the one described in Section 8.2.3 and 8.2.4 is taken to ensure a balanced-force formulation. The derived value at the interface based on the reconstruction is incorporated and the operators are formulated such that the calculated pressure gradients are exact representations of the discontinuity at the interface. The appropriate mesh size is taken in the discrete operators. As an example consider the control volume L with center (i, j) as shown in Figure 8.3. The pressure terms are evaluated as follows:

$$d_2^h p_{\mathbf{i}} = \frac{p_I - \frac{p_{i,j} + p_{i,j-1}}{2}}{\rho_1 H}, \quad (8.32)$$

$$d_1^h p_{\mathbf{i}} = \frac{\frac{p_{i+1,j} + p_{i,j}}{2} - \frac{p_{i,j} + p_{i,j-1}}{2}}{\rho_1 h}, \quad (8.33)$$

$$d_2^{\frac{h}{2}} p_{\mathbf{i} + \frac{1}{2} \hat{e}_2} = \frac{p_I - p_{i,j}}{\rho_1 \frac{H}{2}}. \quad (8.34)$$

8.3.3. RESULT

The operators were implemented on a domain as shown in Figure 8.4. The mesh size normal to the interface is varied by changing the number of subdivisions n_y^2, n_y^1 in direction \hat{e}_2 . A uniform mesh width is employed in direction \hat{e}_1 .

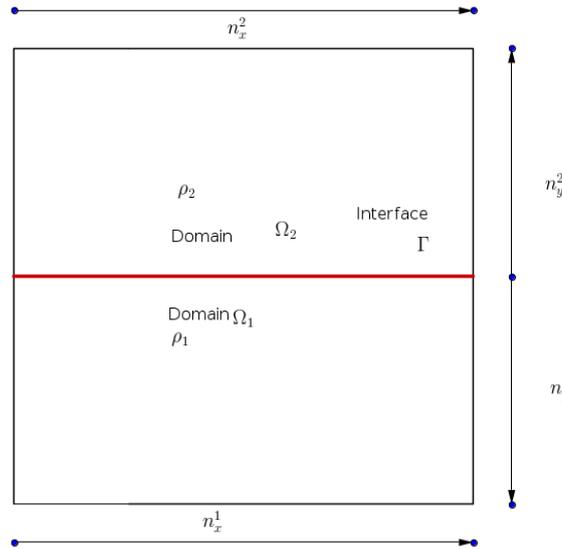


Figure 8.4: 2D Computational non-uniform grid with collocate arrangement of unknowns

Table 8.2 shows the L^2 norm of the residual for a combination of different mesh sizes and density ratios. It can be seen that the residual is independent of the size of the mesh, the non-uniformity and the density ratio.

Table 8.2: L^2 norm of the residual for non-uniform mesh

Density Ratio	Horizontal Mesh Size	Vertical Mesh Size	$\frac{\ DG^f p - r_{DG}\ _2}{N}$	$\frac{\ B^h p - r_B\ _2}{N}$
$\frac{\rho_2}{\rho_1} = 10$	$n_x = 8$	$n_y^2 = 4, n_y^1 = 8$	1.4e-14	9.9e-14
		$n_y^2 = 4, n_y^1 = 16$	1.4e-14	3.2e-14
	$n_x = 16$	$n_y^2 = 4, n_y^1 = 8$	1.5e-14	2.7e-14
		$n_y^2 = 4, n_y^1 = 16$	6.3e-14	5.0e-14

8.4. DISCUSSION

Queutey and Visonneau propose a second order Taylor expansion for the face reconstruction that is based on the *a priori* known discontinuities for multiphase flows [1]. Exact details on the formulation of the pressure-velocity coupling and the operators involved are not provided by the authors. Not only the flow fields have to satisfy the interface jump conditions, but also the operators involved in the discrete system employing the reconstruction should satisfy the interface jump conditions to produce a balanced-force discretization. Hence a formulation of the operators was derived and the results of the implementation were discussed for a uniform and non-uniform grid for quiescent two-phase flow. It was seen that the operators satisfy the analytical piecewise linear profile of the pressure upto machine precision and they produce a discrete force balance.

9

BALANCED-FORCE DISCRETIZATION: NON COINCIDING INTERFACE AND CELL FACE

In this chapter a balanced-force discretization is proposed for the case in which the interface does not coincide with the cell face. First, an extension of the reconstruction approach proposed by Queutey and Visonneau, for an arbitrarily inclined interface based on its location is delineated. Next, the construction and the formulation of the operators are discussed. Then the order of the operators is mathematically derived and numerically verified using the Method of Manufactured Solutions. Finally the reconstruction for velocity at the interface has been outlined, and the results of the implementation are evaluated.

9.1. GENERAL FORMULATION

Consider a general planar interface Γ (shown as the dashed red line in figure 9.1) dividing the computational subdomains Ω^- and Ω^+ . The interface in the limit of the domain Ω^- , Ω^+ is labeled as I^- and I^+ , respectively, and the cell face with the face center F is shown with a solid black line. The vectors \mathbf{l} , \mathbf{r} , \mathbf{f} are the vectors joining the cell center L , R and the face center F to the interface center, respectively. Second order Taylor expansions are employed consecutively to first calculate the value of the scalar at the interface, based on which the face value is calculated using a one sided approximation of the gradient as described below.

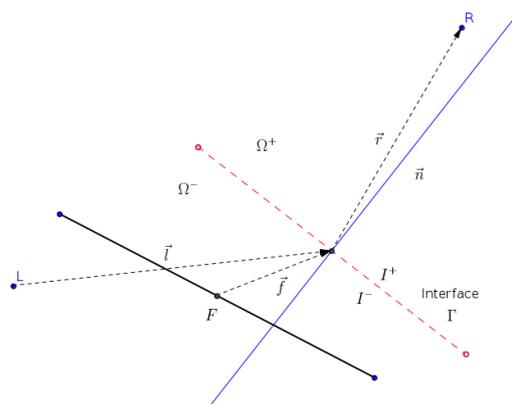


Figure 9.1: Example of a arbitrarily oriented and located interface

Consider a quantity Q , where Q is a scalar field that is continuously differentiable away from the interface. The interface jump conditions at the interface Γ are given by:

$$[Q] = a, \quad (9.1)$$

$$[c\nabla Q] = \mathbf{b}, \quad (9.2)$$

where a , \mathbf{b} and c are known functions. The face and the interface values are calculated as follows:

$$Q_L = Q_{I^-} - \mathbf{l} \cdot \nabla Q^- \quad (9.3)$$

$$Q_R = Q_{I^+} + \mathbf{r} \cdot \nabla Q^+ \quad (9.4)$$

$$Q_F = Q_{I^-} - \mathbf{f} \cdot \nabla Q^- . \quad (9.5)$$

The value at the interface is calculated using equations (9.3) and (9.4) based on the jump conditions (9.1) and (9.2). Next, the value at the face is calculated based on a one sided approximation between the interface and the face using equation (9.5).

9.2. QUIESCENT FLOW: CELL FACE OFFSET AND ALIGNED WITH THE INTERFACE ON A NON-UNIFORM GRID

In this section the balanced-force approach for the case in which the interface is aligned but offset to the interface will be presented. In this case, it is assumed that the gravity is defined as $\mathbf{g} = g\hat{e}_2$, where g is the magnitude of the gravitational force. The pressure reconstruction based on the general formulation presented in section 9.1 is required in the direction \hat{e}_2 to calculate the pressure at the interface and the face. First the pressure reconstruction process is described, followed by description of the construction of operators to achieve a discrete force balance. Finally the results of the implementation are discussed. The test case will help to verify the fidelity of the operators when the general formulation as described in 9.1 is employed.

9.2.1. PRESSURE RECONSTRUCTION AT THE INTERFACE AND THE FACE

Consider the gravity acting as a body force in the direction \hat{e}_2 : the interface is offset to the faces of the control volume as shown in figure 9.2.

$$\mathbf{g} = g\hat{e}_2 \quad (9.6)$$

The pressure reconstruction is required to ensure the discretization satisfies the interface conditions. The interface Γ is offset and aligned with the horizontal face at $i, j + \frac{1}{2}$ by the distance δ .

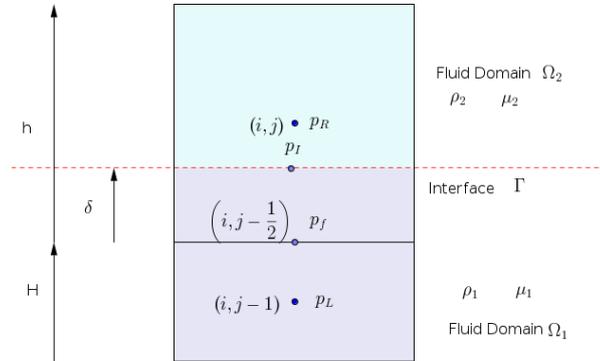


Figure 9.2: Example of offset interface to the face on a 2D collocated grid: discretization for Cell R

The proposed formulation in 9.1 is applied to calculate the pressure as follows:

The pressure field at the interface is subject to

$$[p] = 0, \quad (9.7)$$

$$\left[\frac{\nabla p \cdot \hat{e}_2}{\rho} \right] = 0. \quad (9.8)$$

The values at cell centers L , R are expressed as:

$$p_L = p_I - (H + \delta)\nabla^- p \cdot \hat{e}_2, \quad (9.9)$$

$$p_R = p_I + (h - \delta)\nabla^+ p \cdot \hat{e}_2. \quad (9.10)$$

Hence we get

$$p_I = \frac{P_L(h - \delta)\rho_R + P_R(H + \delta)\rho_I}{(h - \delta)\rho_R + (H + \delta)\rho_I} \quad (9.11)$$

A second order Taylor expansion at point $i, j + \frac{1}{2}$ gives

$$p_f = p_I - \delta(\nabla p^- \cdot \hat{e}_2). \quad (9.12)$$

The projection of the gradient is calculated by a one sided approximation to ensure that the interface jump condition for the discontinuity in the pressure gradient is satisfied.

$$p_f = p_I - \delta \left(\frac{p_I - p_L}{\frac{H}{2} + \delta} \right), \quad (9.13)$$

$$p_f = p_I \left(1 - \frac{\delta}{\frac{H}{2} + \delta} \right) + P_L \left(\frac{\delta}{\frac{H}{2} + \delta} \right). \quad (9.14)$$

As seen above, the value of the pressure at the face is reconstructed based on a one-sided approximation at the interface. For the case in which the interface coincides with the interface, the formulation of the operators DG^f and B , to achieve a discrete force balance, was delineated in Section 8.2.3 and Section 8.2.4. For the case where the interface does not coincide with the faces of the control volumes in the computational grid, a similar approach is taken. The operators are constructed which incorporate the value calculated at the face and measures are taken to ensure gradients that the numerically gradients are directly approximated using the pressure values at either side of the interface.

9.2.2. OPERATOR DG^f

Operator DG^f is defined as

$$DG^f p = h\Delta t \left[\left(d_1^{\frac{h}{2}} p_{\mathbf{i}+\frac{1}{2}\hat{e}_1} - d_1^{\frac{h}{2}} p_{\mathbf{i}-\frac{1}{2}\hat{e}_1} \right) - \left(d_2^{\frac{h}{2}} p_{\mathbf{i}+\frac{1}{2}\hat{e}_2} - d_2^{\frac{h}{2}} p_{\mathbf{i}+\frac{1}{2}\hat{e}_2} \right) \right]. \quad (9.15)$$

When evaluating the operator, a one sided approximation is employed to restrict the calculation of the gradient to one fluid. Further, the local density of the fluid is chosen, which ensures that the \hat{e}_2 component of the operator G^f cancels with gravity.

In the following subsections, the method of discretizing the components of the DG^f operator has been described.

CONTROL VOLUME R

Consider the discretization of the operator in control volume R shown in figure 9.2, the discretization near the interface in the control volume R is employed as follows:

$$d_2^h p_{\mathbf{i}-\frac{1}{2}\hat{e}_2} = \frac{p_f - p_{i,j-\frac{1}{2}}}{\rho_2 \frac{H}{2}}, \quad (9.16)$$

$$d_2^h p_{\mathbf{i}+\frac{1}{2}\hat{e}_2} = \frac{p_{i,j+\frac{1}{2}} - p_{i,j}}{\rho_2 \frac{h}{2}}, \quad (9.17)$$

$$d_1^h p_{\mathbf{i}+\frac{1}{2}\hat{e}_1} = \frac{p_{i+1,j} - p_{i,j}}{\rho_1 \Delta x_1}, \quad (9.18)$$

$$d_1^h p_{\mathbf{i}-\frac{1}{2}\hat{e}_1} = \frac{p_{i,j} - p_{i-1,j}}{\rho_1 \Delta x_1}. \quad (9.19)$$

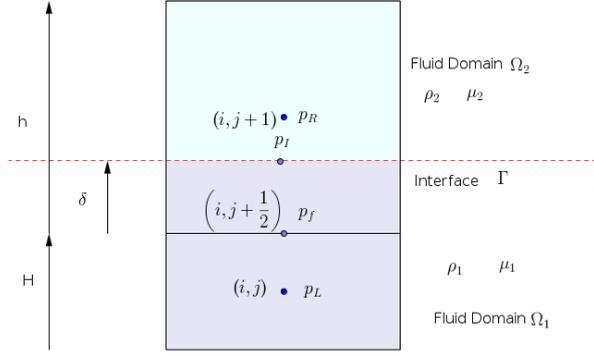


Figure 9.3: Example of offset interface to the face on a 2D collocated grid: discretization for Cell L

CONTROL VOLUME L

Consider the discretization of the operator in control volume R shown in figure 9.3, the discretization of the pressure term in operator DG^f operator in the control volume L is done as follows:

$$d_2^h p_{\mathbf{i} + \frac{1}{2} \hat{\mathbf{e}}_2} = \frac{p_f - p_{i,j}}{\rho_1 \frac{H}{2}}, \quad (9.20)$$

$$d_1^h p_{\mathbf{i} - \frac{1}{2} \hat{\mathbf{e}}_2} = \frac{p_{i,j} - p_{i,j-1}}{\rho_1 H}, \quad (9.21)$$

$$d_1^h p_{\mathbf{i} + \frac{1}{2} \hat{\mathbf{e}}_1} = \frac{p_{i+1,j} - p_{i,j}}{\rho_1 \Delta x_1}, \quad (9.22)$$

$$d_1^h p_{\mathbf{i} - \frac{1}{2} \hat{\mathbf{e}}_1} = \frac{p_{i,j} - p_{i-1,j}}{\rho_1 \Delta x_1}. \quad (9.23)$$

9.2.3. OPERATOR B^c

The operator B^c involves the discretization of the pressure gradients at the center of the cell. For the case in which the interface does not coincide with the cell face, the value of the pressure at the face, reconstructed from the interface jump conditions is incorporated. Further a choice is made between the volumetric averaged density and a numerically defined density, termed as the projected density and denoted by ρ_c , to achieve a discrete force balance, as explained in Section 9.2.4. In the following section the method of discretization is described, followed by the derivation of projected density in Section 9.2.4.

CONTROL VOLUME R

In the presence of an interface in the control volume, such as control volume R shown in Figure 9.2, the pressure derivatives at the center of the cell are discretized as follows:

$$d_1^h p_{\mathbf{i}} = \frac{p_{i+\frac{1}{2},j} - p_{i-\frac{1}{2},j}}{\rho_c \Delta x_1}, \quad (9.24)$$

where ρ_c is the mixture density defined by

$$\rho_c = \frac{\rho_1 \delta + \rho_2 (h - \delta)}{h}. \quad (9.25)$$

The pressure at the cell face is calculated by linear interpolation

$$p_{i+\frac{1}{2},j} = \frac{p_{i+1,j} + p_{i,j}}{2}, \quad (9.26)$$

$$p_{i-\frac{1}{2},j} = \frac{p_{i,j} + p_{i-1,j}}{2}. \quad (9.27)$$

The pressure derivative in the direction \hat{e}_2 at the center of the control volume R is discretized as

$$d_2^h p_{\mathbf{i}} = \frac{p_{i,j+\frac{1}{2}} - p_{i,j-\frac{1}{2}}}{\rho_\xi h}, \quad (9.28)$$

where ρ_ξ is a numerically calculated density referred to as the projected density to achieve a discrete force balance, as explained in section 9.2.4.

CONTROL VOLUME L

On the other hand, when the control volumes are not intersected by the interface, such as for control volume L shown in Figure 9.3, the pressure derivatives at the center of the control volume are discretized as follows:

$$d_1^h p_{\mathbf{i}} = \frac{p_{i+\frac{1}{2},j} - p_{i-\frac{1}{2},j}}{\rho_1 \Delta x_1}, \quad (9.29)$$

$$d_2^h p_{\mathbf{i}} = \frac{p_{i,j+\frac{1}{2}} - p_f}{\rho_1 h}. \quad (9.30)$$

9.2.4. BALANCED-FORCE APPROXIMATION OF THE DENSITY

A prudent choice for the density has to be made when constructing the operators to achieve a force balance, as described in the following subsections.

DISCRETIZATION AT A FACE

1. The pressure gradient is discontinuous at the interface. Hence the discretization at the face is performed by a one-sided approximation to avoid discretization over the discontinuity. Such an approach limits the calculation of the gradient to one fluid, and the density of that fluid is chosen.

For the control volume R shown in figure 9.3, we have

$$d_1^h p_{\mathbf{i}+\frac{1}{2}\hat{e}_1} = \frac{p_{i+1,j} - p_{i,j}}{\rho_2 \Delta x_1}, \quad (9.31)$$

$$d_2^h p_{\mathbf{i}+\frac{1}{2}\hat{e}_2} = \frac{p_{i,j+1} - p_{i,j}}{\rho_2 h}. \quad (9.32)$$

Similarly,

$$d_2^h p_{\mathbf{i}-\frac{1}{2}\hat{e}_2} = \frac{p_f - p_{i,j-1}}{\rho_1 h}. \quad (9.33)$$

DISCRETIZATION AT A CELL CENTER

When evaluating the pressure gradients at the center of the control volumes, a numerically constructed density has to be used, termed as the projected density. In the following subsection the reasoning and the formulation of projected density has been presented.

1. Derivative in the \hat{e}_2 direction

Consider the control volume R in figure 9.2. A fraction of the control volume is occupied by each fluid, hence the density is chosen such that the projection of the gradient of the pressure in the operator B^c in direction \hat{e}_2 cancels out with the gravity as follows:

$$p_{i,j+\frac{1}{2}} - p_I = \rho_2 g(h - \delta), \quad (9.34)$$

$$p_I - p_{i,j-\frac{1}{2}} = \rho_1 g\delta. \quad (9.35)$$

Adding (9.35) and (9.34) gives

$$p_{i,j+\frac{1}{2}} - p_{i,j-\frac{1}{2}} = \rho_2 g(h - \delta) + \rho_1 g \delta \quad (9.36)$$

Dividing by h on both sides gives

$$\frac{p_{i,j+\frac{1}{2}} - p_{i,j-\frac{1}{2}}}{h} = \left(\rho_2 \frac{h - \delta}{h} + \rho_1 \frac{\delta}{h} \right) g. \quad (9.37)$$

From equation (9.37) an equivalent density, referred to as the 'projected' density is defined as

$$\rho_\xi = \left(\rho_2 \frac{h - \delta}{h} + \rho_1 \frac{\delta}{h} \right) = \rho_2 \phi_{d_{i,j-\frac{1}{2}}} + \rho_1 \phi_{d_{i,j+\frac{1}{2}}}, \quad (9.38)$$

where ϕ_ξ is the projection of the level set field (ϕ) in direction \hat{e}_2 .

Hence,

$$d_2^h p_1 = \frac{p_{i,j+\frac{1}{2}} - p_{i,j-\frac{1}{2}}}{\rho_\xi h}. \quad (9.39)$$

2. Derivative in the \hat{e}_1 direction

The gravity does not contribute to the pressure gradient in the direction \hat{e}_1

$$\mathbf{g} \cdot \hat{e}_1 = 0. \quad (9.40)$$

There is no explicit choice for the density when evaluating component \hat{e}_1 of the pressure gradient. Hence the volumetric averaged density is chosen when calculating the \hat{e}_1 component of the pressure.

It is clear that the discussed approach produces a discrete force balance as the discretization satisfies the interface jump conditions by employing a pressure reconstruction and the projected density.

9.2.5. RESULT

The operators were implemented and tested for uniform and non-uniform grids. Table 9.1 shows the L^2 norm of the residuals of the discrete operators B , DG^f . It is evident that the formulation is independent of the density ratio, mesh size, non-uniformity and location of the interface.

Table 9.1: L^2 norm of the residuals for the operators DG^f , B

Density Ratio	$\frac{\delta}{h}$	Horizontal Mesh Size	Vertical Mesh Size	$\frac{\ DG^f p - r_{DG}\ _2}{N}$	$\frac{\ B^h p - r_B\ _2}{N}$
$\frac{\rho_1}{\rho_2} = 1$	0.2	$n_x^1 = n_x^2 = 8$	$n_y^2 = 4, n_y^1 = 8$	1.8e-14	7.8e-15
			$n_y^2 = 4, n_y^1 = 16$	4.3e-14	6.9e-14
		$n_x^1 = n_x^2 = 16$	$n_y^2 = 4, n_y^1 = 8$	5.1e-14	5.4e-15
			$n_y^2 = 4, n_y^1 = 16$	8.2e-14	1.5e-14
$\frac{\rho_1}{\rho_2} = 1000$	0.2	$n_x^1 = n_x^2 = 8$	$n_y^2 = 4, n_y^1 = 8$	7.9e-14	9.3e-15
			$n_y^2 = 4, n_y^1 = 16$	2.5e-13	2.0e-14
		$n_x^1 = n_x^2 = 16$	$n_y^2 = 4, n_y^1 = 8$	7.4e-14	1.8e-14
			$n_y^2 = 4, n_y^1 = 16$	1.5e-14	5.4e-15
	0.7	$n_x^1 = n_x^2 = 8$	$n_y^2 = 4, n_y^1 = 8$	6.1e-14	1.1e-14
			$n_y^2 = 4, n_y^1 = 16$	1.9e-14	1.4e-14
		$n_x^1 = n_x^2 = 16$	$n_y^2 = 4, n_y^1 = 8$	1.4e-14	9.2e-15
			$n_y^2 = 4, n_y^1 = 16$	4.0e-14	1.5e-14

9.3. INCLINED INTERFACE

Consider an inclined interface Γ , as represented by the red dotted line in Figure 9.4. The gravity is assumed to be directed normal to the interface. Reconstruction is required in both Cartesian directions as gravity contributes to the vector projections of the pressure gradient in both directions \hat{e}_1 and \hat{e}_2 . For an arbitrarily inclined interface the distance to the interface varies in each control volume, which requires a reconstruction of the face according to the distance to the interface. In the following sections, first the pressure reconstruction for a arbitrarily inclined interface is discussed, followed by construction of the operators and discussion of results of the implementation.

9.3.1. PRESSURE RECONSTRUCTION AT THE FACE AND THE INTERFACE

The interface divides the computational domain into the subdomains Ω_1 and Ω_2 . The pressure at the intersection of line segment joining cell centers $i, j+1$ and i, j is denoted by p_I , while the pressure at the center of the face is referred to as p_f .

The gravity is normal to the interface and given by

$$\mathbf{g} = g\mathbf{n}. \quad (9.41)$$

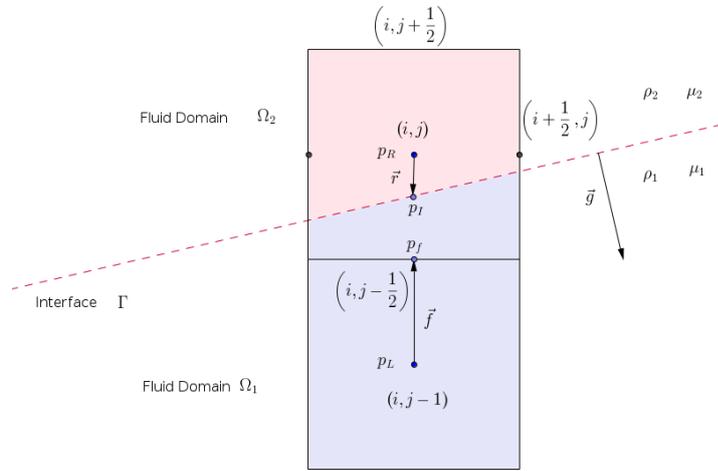


Figure 9.4: Example of an inclined interface : Face reconstruction from cell center below the interface

PRESSURE AT THE INTERFACE

Retaining a second order Taylor expansion at points R and L gives

$$P_R = p_I + \mathbf{r} \cdot \nabla_I p, \quad (9.42)$$

$$p_L = p_I - \mathbf{l} \cdot \nabla_I p. \quad (9.43)$$

Decomposing the vectors \mathbf{l} , \mathbf{r} in directions normal and tangential to the interface gives

$$P_R = p_I + (\mathbf{r}_n + \mathbf{r}_t) \cdot \nabla_I p, \quad (9.44)$$

$$p_L = p_I - (\mathbf{l}_n + \mathbf{l}_t) \cdot \nabla_I p. \quad (9.45)$$

The tangential vector projection of the pressure gradient is zero because the gravity acts normal to the interface, hence

$$\mathbf{r}_t \cdot \nabla_I p = 0, \quad (9.46)$$

$$\mathbf{l}_t \cdot \nabla_I p = 0. \quad (9.47)$$

Equations (9.44) and (9.45) simplify to

$$p_R = p_I + \mathbf{r}_n \cdot \nabla_I p, \quad (9.48)$$

$$p_L = p_I - \mathbf{l}_n \cdot \nabla_I p. \quad (9.49)$$

or

$$p_R = p_I + r_n \mathbf{n} \cdot \nabla_I p, \quad (9.50)$$

$$p_L = p_I - l_n \mathbf{n} \cdot \nabla_I p, \quad (9.51)$$

where, r_n, l_n are the distances from points R and L to the interface, respectively. Further

$$\left[\frac{\nabla p \cdot \mathbf{g}}{\rho} \right] = 0. \quad (9.52)$$

Multiplying equation (9.50) by $\frac{1}{\rho_1 r_n}$, and equation (9.51) by $\frac{1}{\rho_2 l_n}$ gives

$$p_I = p_L \frac{\rho_2 r_n}{\rho_1 l_n + \rho_2 r_n} + p_R \frac{\rho_1 l_n}{\rho_1 l_n + \rho_2 r_n}. \quad (9.53)$$

PRESSURE AT THE FACE CENTER

The value of the pressure at the face is calculated using a Taylor expansion using a one-sided approximation as explained in the following section.

Face value reconstruction from cell center below the interface

Consider the interface between the face and cell center center R as shown in the figure 9.4. Retaining a second order Taylor expansion

$$p_f = p_L + \mathbf{f} \cdot \nabla_L p, \quad (9.54)$$

$$\mathbf{f} = f \hat{\mathbf{e}}_2 \cdot \nabla_L p. \quad (9.55)$$

The projection of the pressure gradient at L is calculated by

$$\hat{\mathbf{e}}_2 \cdot \nabla_L p = \frac{p_I - p_L}{l}. \quad (9.56)$$

Hence

$$p_f = p_L + \frac{f}{l} (p_I - p_L), \quad (9.57)$$

$$\begin{aligned} p_f &= p_L + \frac{f}{l} \left(p_L \frac{\rho_2 r_n}{\rho_1 l_n + \rho_2 r_n} + p_R \frac{\rho_1 l_n}{\rho_1 l_n + \rho_2 r_n} - p_L \right) \\ &= \left(1 - \frac{f}{l} \right) p_L + \left(\frac{f}{l} \right) \left(\frac{\rho_2 r_n}{\rho_1 l_n + \rho_2 r_n} \right) p_L + \left(\frac{f}{l} \right) \left(\frac{\rho_1 l_n}{\rho_1 l_n + \rho_2 r_n} \right) p_R. \end{aligned} \quad (9.58)$$

Face value reconstruction from cell center above the interface

Consider the interface between the face and cell center L as shown in figure 9.5, the value at the face is constructed from gradient at point R , which gives

$$p_f = p_R - \mathbf{f} \cdot \nabla_R p, \quad (9.59)$$

$$p_f = p_R - \frac{f}{l} (p_R - p_I),$$

$$\begin{aligned} p_f &= p_R - \frac{f}{l} \left(p_R - p_L \frac{\rho_2 r_n}{\rho_1 l_n + \rho_2 r_n} + p_R \frac{\rho_1 l_n}{\rho_1 l_n + \rho_2 r_n} \right), \\ &= \left(1 - \frac{f}{l} \right) p_R + \left(\frac{f}{l} \right) \left(\frac{\rho_2 r_n}{\rho_1 l_n + \rho_2 r_n} \right) p_L + \left(\frac{f}{l} \right) \left(\frac{\rho_1 l_n}{\rho_1 l_n + \rho_2 r_n} \right) p_R. \end{aligned} \quad (9.60)$$

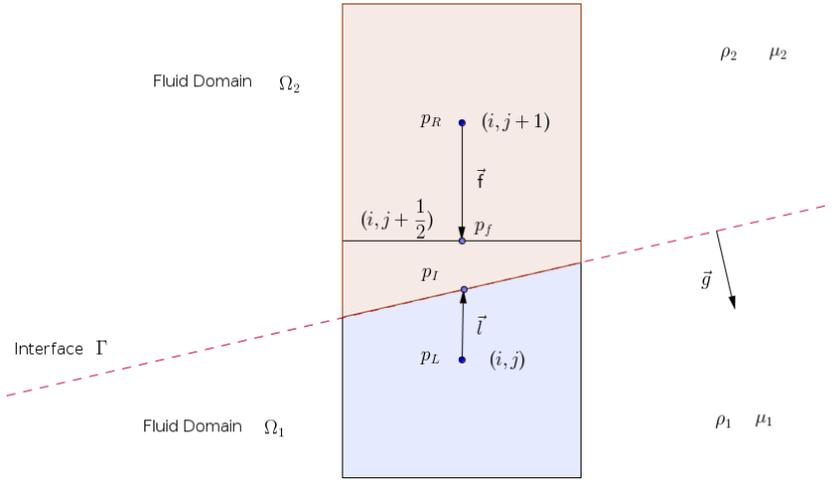


Figure 9.5: Example of an inclined interface: Face value reconstruction from cell center from above the interface

RECONSTRUCTION IN DIRECTION \hat{e}_1

The face reconstruction at the vertical faces of the control volumes is carried out in a similar fashion. Figures 9.6 and 9.7 show examples where a reconstruction is needed in direction \hat{e}_1 .

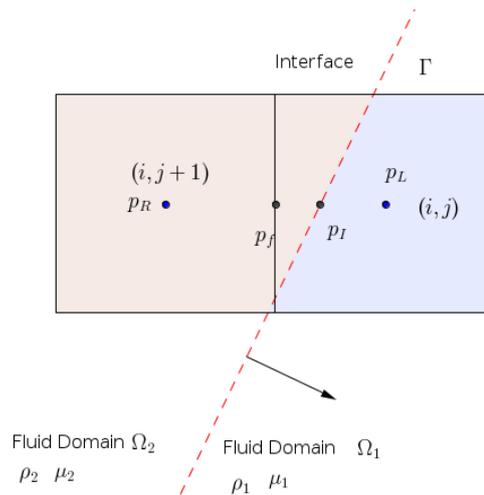


Figure 9.6: Example of an inclined interface: Face value reconstruction from cell center to the left of the interface

9.3.2. BALANCED-FORCE APPROXIMATION OF THE DENSITY

As explained in section 9.2.4, when evaluating the pressure gradients at the face, a one sided approximation is employed, and the local density at the face is chosen to consistently evaluate the operators. When evaluating the pressure gradient at the cell center the projected density is utilized to ensure a discrete force balance between the pressure gradient and the gravity.

$$\nabla p \cdot \mathbf{n} = \rho \mathbf{g} \cdot \mathbf{n} \tag{9.61}$$

In the case presented in section 9.3 the gravity contributes to the pressure gradient in both the coordinate directions (9.61). Hence the projected density has to be employed when evaluating both the components of operator B^c . The level set field is projected in the direction \hat{e}_1 and \hat{e}_2 when evaluating the \hat{e}_1 and \hat{e}_2 component of operator B^c , respectively.

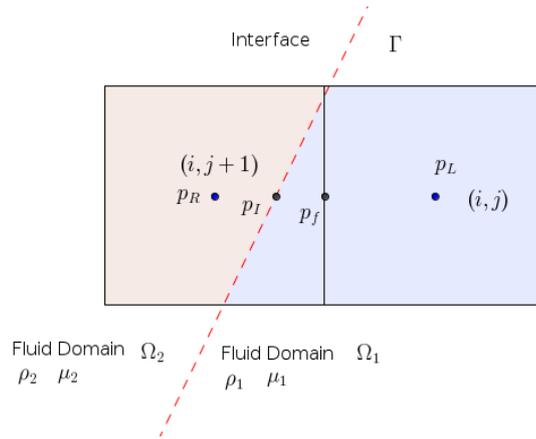


Figure 9.7: Example of an inclined interface: Face value reconstruction from cell center to the right of the interface

Consider the discretization of the pressure gradient in the control volume R as shown in figure 9.4.

DISCRETIZATION AT A FACE

When evaluating the gradients of the pressure at the face, the local density, i.e. the density of the fluid present at the face, is chosen.

DISCRETIZATION AT A CELL CENTER

Consider evaluation of $d_2^h p_R$ in control volume R . The following relation between pressure hold

$$p_{i,j+\frac{1}{2}} - p_I = \rho_2 g \xi_{i,j+\frac{1}{2}}^2, \quad (9.62)$$

$$p_I - p_f = \rho_1 g \xi_f, \quad (9.63)$$

where, ξ^2 represents the projection of the level set field in direction \hat{e}_2 .

Adding equation 9.62 and 9.63 gives

$$p_{i,j+\frac{1}{2}} - p_f = \left(\rho_2 \xi_{i,j+\frac{1}{2}}^2 + \rho_1 \xi_f \right) g. \quad (9.64)$$

Dividing both sides by h we get

$$\frac{1}{\frac{\left(\rho_2 \xi_{i,j+\frac{1}{2}}^2 + \rho_1 \xi_f^2 \right)}{h}} \frac{p_{i,j+\frac{1}{2}} - p_f}{h} = g. \quad (9.65)$$

$$\frac{1}{\rho_{\xi}^R} \frac{p_{i,j+\frac{1}{2}} - p_f}{h} = g, \quad (9.66)$$

where, ρ_{ξ}^R represents the density in cell R weighted by the projected distance from points $i, j + \frac{1}{2}$ and f .

$$\rho_{\xi}^R = \frac{\rho_2 \xi_{i,j+\frac{1}{2}}^2 + \rho_1 \xi_f^2}{h} \quad (9.67)$$

It is clear from equations 9.66 and 9.65 that proposed ensures that the pressure gradient calculated as above satisfies a discrete force balance with the gravity.

Hence,

$$d_2^h p_R = \frac{p_{i,j+\frac{1}{2}} - p_f}{\rho_{\xi}^R h} \quad (9.68)$$

Similarly when evaluating the scalar projection of the pressure gradient in \hat{e}_1 direction, the projected density ρ_{ξ}^1 is employed to achieve a discrete force balance.

9.3.3. DISCRETE OPERATORS

In the following Section a force balanced formulation for the arbitrarily inclined interface is presented. As seen in Section 8.2.1 and 9.2, the operators are constructed by:

- Employing the reconstructed value at the face based on the interface jump conditions.
- Using a one sided approximation.
- Making an appropriate choice of the density between the volumetric averaged density, the projected density and the local density of the fluid to achieve a discrete force balance.

In the following Subsections, the method of discretization, taking the above delineated measures to achieve a discrete force balance is described for the case of a planar interface not aligned with or coinciding with the coordinate directions.

DG^f

Consider the discretization of the operator DG in the control volume R , as shown in figure 9.4. A one sided discretization is carried out as follows

$$d_2^h p_{\mathbf{i}-\frac{1}{2}\hat{e}_2} = \frac{p_f - p_{i,j-1}}{\rho_1 h}, \quad (9.69)$$

where p_f is the pressure calculated by the face value reconstruction, as explained in section 9.3.1.

$$d_2^h p_{\mathbf{i}+\frac{1}{2}\hat{e}_2} = \frac{p_{i,j+1} - p_{i,j}}{\rho_2 h} \quad (9.70)$$

$$d_1^h p_{\mathbf{i}+\frac{1}{2}\hat{e}_1} = \frac{p_{i+1,j} - p_{i,j}}{\rho_2 h} \quad (9.71)$$

$$d_1^h p_{\mathbf{i}-\frac{1}{2}\hat{e}_1} = \frac{p_{i,j} - p_{i-1,j}}{\rho_2 h} \quad (9.72)$$

B

1. Component of pressure gradient in direction \hat{e}_2 .

Consider the discretization of the pressure gradient in the control volume R , as shown in figure 9.4. The evaluation of the component in direction \hat{e}_2 is done as follows:

$$d_2^h p_{\mathbf{i}} = \frac{p_{i,j+\frac{1}{2}} - p_f}{\rho_{\xi}^R h}, \quad (9.73)$$

in which, the pressure at $i, j + \frac{1}{2}$ is evaluated by linear interpolation from the adjacent cell values,

$$d_2^h p_{\mathbf{i}} = \frac{p_{i,j+1} + p_{i,j}}{2}. \quad (9.74)$$

As explained in section 9.3.2, the projected density is utilized.

2. Component of pressure gradient in direction \hat{e}_1

The \hat{e}_1 component of the pressure gradient in control volume R , as shown in figure 9.4, is evaluated as

$$d_1^h p_{\mathbf{i}} = \frac{p_{i+\frac{1}{2},j} - p_{i-\frac{1}{2},j}}{\rho_2 h}. \quad (9.75)$$

When the interface intersects the vertical edges of the control volumes, as shown in figures 9.6 and 9.7, a approach similar to (9.73) is taken by employing the projected distance ρ_{ξ}^1 density in the direction \hat{e}_1 .

9.3.4. BOUNDARY CONDITION

Inhomogeneous Neumann boundary conditions are imposed to concur with

$$\nabla p \cdot \mathbf{n}_b = \mathbf{g} \cdot \mathbf{n}_b, \quad (9.76)$$

where \mathbf{n}_b is the vector normal to the interface.

9.3.5. RESULTS

Table 9.2 shows the L^2 norm of the residuals for the implementation of the balanced-force formulation near an inclined interface for different slopes, density ratios and mesh widths. It can be seen that the residuals are up to machine precision, which shows a successful reconstruction for an arbitrary interface based on the general formulation and a discrete force balance.

Table 9.2: L^2 norm of the residuals for operators DG^f and B

Density Ratio	Slope	x_2 intercept	Mesh Size	$\frac{\ DG^f p - r_{DG}\ _2}{N}$	$\frac{\ B^h p - r_B\ _2}{N}$
$\frac{\rho_1}{\rho_2} = 1$	0.5	0.5	32	2.7e-15	2.3e-15
$\frac{\rho_1}{\rho_2} = 1000$	0.5	0.5	8	1.3e-15	1.2e-15
			16	1.2e-15	1.1e-15
			32	1.6e-15	1.3e-15
	0.7	0.1	8	1.9e-15	1.4e-15
			16	1.7e-15	1.6e-15
			32	1.6e-15	1.8e-15

9.4. METHOD OF MANUFACTURED SOLUTIONS

The B operator constructed in the pressure velocity coupling suppresses checker-board modes and strongly couples the pressure-velocity field. To ascertain that the operator behave as expected, the Method of Manufactured Solutions is used to numerically calculate the order of the operators, which is compared with the mathematically calculated order. First the order of the operators is derived and thereafter the results of the Method of Manufactured Solutions is discussed.

DEFINITION OF THE LOCAL TRUNCATION ERROR

Consider a differential equation defined as

$$L\phi = q, \quad (9.77)$$

where L is a differential operator, ϕ a sufficiently differentiable scalar field and q a known source term.

The discrete approximation of the differential equation is given by

$$L^h \phi_i = q_i. \quad (9.78)$$

The local truncation error is the difference between the exact solution to the differential equation (9.77) and the numerical solution (9.78), i.e.

$$\tau_i = L^h \phi(x_i) - L^h \phi_i, \quad (9.79)$$

where, $\phi(x_i)$ is the exact solution to the differential equation, and ϕ_i the discrete solution.

9.4.1. CONTINUOUS FORM OF DISCRETE OPERATORS

DG^f

The operator DG^f is an approximation of the continuous operator DG_c^f defined as:

$$DG_c^f p(\mathbf{x}_i) := \int_{\Omega_i} \frac{1}{\rho} \nabla \cdot \nabla p d\Omega \quad (9.80)$$

and the source term can be defined as $q(\mathbf{x}_i) := DG_c^f p(\mathbf{x}_i)$.

A pressure field $p(\mathbf{x}_i)$ is chosen such that $DG_c^f p(\mathbf{x}_i) \neq 0$, $x \in \Omega_i$ and such that the pressure field complies with the interface and the boundary conditions with which $q(\mathbf{x}_i)$ is evaluated. The pressure field is now the solution of

$$DG_c^f \phi(\mathbf{x}_i) = q(\mathbf{x}_i). \quad (9.81)$$

The discrete approximation of (9.81) is given by

$$DG^f \phi_i = q(\mathbf{x}_i). \quad (9.82)$$

Hence, the local truncation error is given by:

$$\tau_i = DG^f \phi(\mathbf{x}_i) - DG_c^f \phi(\mathbf{x}_i) = DG^f \phi(\mathbf{x}_i) - \int_{\Omega_i} \frac{1}{\rho} \nabla \cdot \nabla \phi d\Omega. \quad (9.83)$$

EXPECTED ORDER OF ACCURACY

Away from the interface

Consider the evaluation of the operator DG^f away from the interface

$$DG^f p(\mathbf{x}_i) = \frac{h}{\rho} \left(p_{,2} \Big|_{\mathbf{i} + \frac{1}{2} \hat{\mathbf{e}}_2} - p_{,2} \Big|_{\mathbf{i} - \frac{1}{2} \hat{\mathbf{e}}_2} \right) + \frac{h}{\rho} \left(p_{,1} \Big|_{\mathbf{i} + \frac{1}{2} \hat{\mathbf{e}}_1} - p_{,1} \Big|_{\mathbf{i} - \frac{1}{2} \hat{\mathbf{e}}_2} \right) \quad (9.84)$$

Using a Taylor expansion around \mathbf{i} gives

$$\begin{aligned} DG^f p(\mathbf{x}_i) &= \frac{h}{\rho} \left(p'_{\mathbf{i}} + (0.5h) p''_{\mathbf{i} \hat{\mathbf{e}}_\alpha} + \frac{(0.5h)^2}{2!} p'''_{\mathbf{i} \hat{\mathbf{e}}_\alpha} + \frac{(0.5h)^3}{3!} p''''_{\mathbf{i} \hat{\mathbf{e}}_\alpha} + \frac{(0.5h)^4}{4!} p''''''_{\mathbf{i} \hat{\mathbf{e}}_\alpha} + \dots \right) \\ &\quad - \frac{h}{\rho} \left(p'_{\mathbf{i}} - (0.5h) p''_{\mathbf{i} \hat{\mathbf{e}}_\alpha} + \frac{(0.5h)^2}{2!} p'''_{\mathbf{i} \hat{\mathbf{e}}_\alpha} - \frac{(0.5h)^3}{3!} p''''_{\mathbf{i} \hat{\mathbf{e}}_\alpha} + \frac{(0.5h)^4}{4!} p''''''_{\mathbf{i} \hat{\mathbf{e}}_\alpha} + \dots \right). \end{aligned} \quad (9.85)$$

Simplifying this gives

$$DG^f p(\mathbf{x}_i) = \frac{1}{\rho} \left(h^2 p''_{\mathbf{i} \hat{\mathbf{e}}_\alpha} + \frac{h^4}{24} p''''_{\mathbf{i} \hat{\mathbf{e}}_\alpha} \right) + \mathcal{O}(h^6). \quad (9.86)$$

Now, the discrete operator $DG^f p_i$ is given by

$$DG^f p_i = h \left(\frac{p_{\mathbf{i} + \hat{\mathbf{e}}_\alpha} - p_{\mathbf{i}}}{\rho h} - \frac{p_{\mathbf{i}} - p_{\mathbf{i} - \hat{\mathbf{e}}_\alpha}}{\rho h} \right). \quad (9.87)$$

Using a Taylor expansion around \mathbf{i} gives

$$\begin{aligned} DG^f p_i &= \frac{1}{\rho} \left(h p'_{\mathbf{i} \hat{\mathbf{e}}_\alpha} + \frac{h^2}{2!} p''_{\mathbf{i} \hat{\mathbf{e}}_\alpha} + \frac{h^3}{3!} p'''_{\mathbf{i} \hat{\mathbf{e}}_\alpha} + \frac{h^4}{4!} p''''_{\mathbf{i} \hat{\mathbf{e}}_\alpha} + \dots \right) - \frac{1}{\rho} \left(h p'_{\mathbf{i} \hat{\mathbf{e}}_\alpha} - \frac{h^2}{2!} p''_{\mathbf{i} \hat{\mathbf{e}}_\alpha} + \frac{h^3}{3!} p'''_{\mathbf{i} \hat{\mathbf{e}}_\alpha} - \frac{h^4}{4!} p''''_{\mathbf{i} \hat{\mathbf{e}}_\alpha} + \dots \right), \\ &= \frac{1}{\rho} \left(h^2 p''_{\mathbf{i} \hat{\mathbf{e}}_\alpha} + \frac{h^4}{12} p''''_{\mathbf{i} \hat{\mathbf{e}}_\alpha} \right) + \mathcal{O}(h^6). \end{aligned} \quad (9.88)$$

Hence,

$$DG^f p_i = \frac{1}{\rho} \left(h^2 p''_{\mathbf{i} \hat{\mathbf{e}}_\alpha} + \frac{h^4}{12} p''''_{\mathbf{i} \hat{\mathbf{e}}_\alpha} \right) + \mathcal{O}(h^6). \quad (9.89)$$

Subtracting the exact solution to the differential equation (9.86) from the discrete solution (9.89) gives

$$\tau_i^{DG^f} = \frac{h^4}{12\rho} p''''_{\mathbf{i} \hat{\mathbf{e}}_\alpha} + \mathcal{O}(h^6). \quad (9.90)$$

Hence, it can be seen the local truncation error of the operator DG^f is fourth order accurate at a sufficiently large distance from the interface.

Near the interface

Consider the case where the interface coincides with the cell face at the top face of the control volume as shown in figure 8.1. The pressure at $\mathbf{i} + \frac{1}{2}\hat{e}_2$ is calculated by the pressure reconstruction, as explained in section 8.2.1, which gives

$$p_{\mathbf{i}+\frac{1}{2}\hat{e}_2} = p_{\mathbf{i}} + hp'_{\mathbf{i}\hat{e}_2} \quad (9.91)$$

A standard discretization is employed for component \hat{e}_1 of operator DG^f . As seen in equation 9.90, the local truncation error will be of the order 4. On the other hand, the \hat{e}_2 component incorporates the reconstruction to achieve a discrete force balanced given by

$$DG^f \hat{e}_2 p_{\mathbf{i}} = \frac{1}{\rho} \left(p_{\mathbf{i}} + hp'_{\mathbf{i}+\frac{1}{2}\hat{e}_2} - p_{\mathbf{i}} \right) - \frac{1}{\rho} \left(p_{\mathbf{i}} - p_{\mathbf{i}-\frac{1}{2}\hat{e}_2} \right). \quad (9.92)$$

Employing a Taylor expansion at point \mathbf{i} gives

$$\begin{aligned} DG^f \hat{e}_2 p_{\mathbf{i}} &= \frac{1}{\rho} \left(p_{\mathbf{i}} + hp'_{\mathbf{i}+\frac{1}{2}\hat{e}_2} - p_{\mathbf{i}} \right) - \frac{1}{\rho} \left(p_{\mathbf{i}} - \left(p_{\mathbf{i}} - hp'_{\mathbf{i}\hat{e}_2} + h^2 p''_{\mathbf{i}\hat{e}_2} - h^3 p'''_{\mathbf{i}\hat{e}_2} + \dots \right) \right), \\ &= \frac{1}{\rho} \left(h^2 p''_{\mathbf{i}+\frac{1}{2}\hat{e}_2} - h^3 p'''_{\mathbf{i}+\frac{1}{2}\hat{e}_2} \right) + \mathcal{O}(h^4). \end{aligned} \quad (9.93)$$

Subtracting the exact solution to the differential equation (9.86) from the discrete solution (9.93) gives

$$\tau_{\mathbf{i}}^{DG^f} = \frac{h^2}{\rho} p''_{\mathbf{i}+\frac{1}{2}\hat{e}_2} + \mathcal{O}(h^3). \quad (9.94)$$

It can be seen that the local truncation error of the operator is DG^f is of the order 2 near the interface.

B OPERATOR

Away from the interface

The discrete operator utilized in the pressure-velocity coupling, $\dot{B}p_{\mathbf{i}}$, is given as

$$\dot{B}^\alpha p_{\mathbf{i}} = \frac{1}{\rho} \left[\frac{1}{2} \left(p_{,\alpha \mathbf{i}} + p_{,\alpha \mathbf{i}+\hat{e}_\alpha} \right) - p_{,\alpha \mathbf{i}+\frac{1}{2}\hat{e}_\alpha} \right]. \quad (9.95)$$

In this subsection, the continuous differential counterpart of the operator is derived. Next, the order of the local truncation error is inferred by subtracting the exact differential solution from the discrete solution.

Employing a Taylor expansion around $\mathbf{i}_\alpha + \frac{1}{2}\hat{e}_\alpha$ gives

$$p'_{\mathbf{i}\alpha} = p'_{\mathbf{i}+\frac{1}{2}\hat{e}_\alpha \alpha} - \frac{h}{2} p''_{\mathbf{i}+\frac{1}{2}\hat{e}_\alpha \alpha} + \frac{h^2}{8} p'''_{\mathbf{i}+\frac{1}{2}\hat{e}_\alpha \alpha} - \frac{h^3}{48} p''''_{\mathbf{i}+\frac{1}{2}\hat{e}_\alpha \alpha} \dots \quad (9.96)$$

$$p'_{\mathbf{i}+\hat{e}_\alpha \alpha} = p'_{\mathbf{i}+\frac{1}{2}\hat{e}_\alpha \alpha} + \frac{h}{2} p''_{\mathbf{i}+\frac{1}{2}\hat{e}_\alpha \alpha} + \frac{h^2}{8} p'''_{\mathbf{i}+\frac{1}{2}\hat{e}_\alpha \alpha} + \frac{h^3}{48} p''''_{\mathbf{i}+\frac{1}{2}\hat{e}_\alpha \alpha} \dots \quad (9.97)$$

Adding equations (9.96) and (9.97) we get

$$p'_{\mathbf{i}\alpha} + p'_{\mathbf{i}+\hat{e}_\alpha \alpha} = 2p'_{\mathbf{i}+\frac{1}{2}\hat{e}_\alpha \alpha} + \frac{h^2}{4} p'''_{\mathbf{i}+\frac{\hat{e}_\alpha}{2} \alpha} + \mathcal{O}(h^4). \quad (9.98)$$

Rewriting equation (9.98) gives

$$\frac{1}{\rho} \left(\frac{1}{2} \left(p'_{\mathbf{i}\alpha} + p'_{\mathbf{i}+\hat{e}_\alpha \alpha} \right) - p'_{\mathbf{i}+\frac{1}{2}\hat{e}_\alpha \alpha} \right) + \mathcal{O}(h^4) = \frac{h^2}{8} p'''_{\mathbf{i}+\frac{\hat{e}_\alpha}{2} \alpha}. \quad (9.99)$$

Assuming $\nabla p \cdot \hat{e}_1 = 0$ gives

$$p''' = (\nabla \nabla \cdot \nabla p). \quad (9.100)$$

From equations (9.100) and (9.99)

$$\dot{B}^\alpha p_{\mathbf{i}} + \mathcal{O}(h^4) = (\nabla \nabla \cdot \nabla p). \quad (9.101)$$

Taking the divergence of equation (9.101) and integrating gives

$$\int \nabla \cdot (\nabla \nabla \cdot \nabla p) \Omega - D \dot{B}^\alpha p_{\mathbf{i}} = \int \nabla \cdot (\nabla \nabla \cdot \nabla p) \Omega - B P_{\mathbf{i}} = \mathcal{O}(h^4). \quad (9.102)$$

Hence,

$$\tau_i^B = \mathcal{O}(h^4). \quad (9.103)$$

Near the interface

A similar numerical analysis can be done for operator B to evaluate the order of the local truncation error near the interface, as done in section 9.4.1. The operator can be split as

$$B = B^f + B^c = DG^f - DG^c. \quad (9.104)$$

The approximation of the first term in (9.104) is of the order $\mathcal{O}(h^2)$, hence a higher order may not be expected, due to the fact the local truncation error is of the order 2 for the first term.

9.4.2. CONVERGENCE OF DISCRETE OPERATORS

Equations (9.105) and (9.106) describe the discrete formulation and its continuous counterpart. A known function is used in the Method of Manufactured Solutions to determine the order of a numerical method by evaluating the convergence characteristics between the discrete and the continuous form (source term) as the computational grid is refined:

$$DGp = \int \frac{1}{\rho} \nabla \cdot \nabla p d\Omega, \quad (9.105)$$

$$Bp = \frac{h^2}{8} \int \frac{1}{\rho} \nabla \cdot (\nabla \nabla \cdot \nabla p) d\Omega. \quad (9.106)$$

Consider Figure 9.13 in which ϵ is the distance from the origin O to the interface.

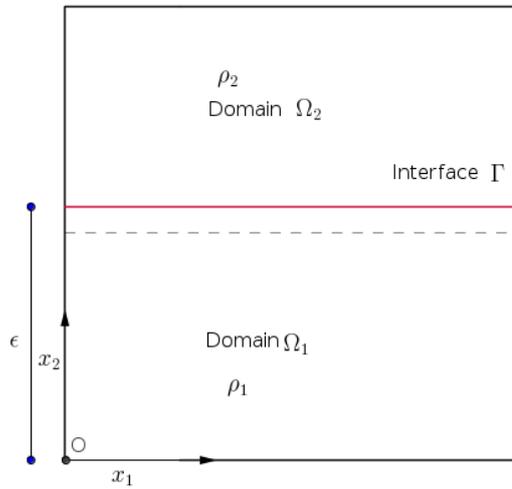


Figure 9.8: Computational Domain utilized in the Method of Manufactured Solutions

The function utilized in the method should satisfy the boundary and interface conditions. In this case, the solution to the operators is subject to

$$\frac{1}{\rho} p^{\Omega_1, 2}(x_1, 0) = g, \quad (9.107)$$

$$p^{\Omega_2}(x_1, 1) = 0, \quad (9.108)$$

$$p^{\Omega_1}(x_1, \epsilon) = p^{\Omega_2}(x_1, \epsilon), \quad (9.109)$$

$$\left[\frac{\nabla p(x_1, \epsilon) \cdot \hat{\mathbf{e}}_2}{\rho(\mathbf{x})} \right] = 0, \quad (9.110)$$

where the density $(\rho(\mathbf{x}))$, is a function of the location in the computational domain \mathbf{x} .

A fourth order polynomial was constructed that satisfy conditions (9.107)-(9.110) as follows

$$p^{\Omega_1} = \rho_1 g x_2^4 + \rho_1 g x_2 - ((\rho_1 - \rho_2)(\epsilon^4 + \epsilon)g + 2\rho_2 g), \quad (9.111)$$

$$p^{\Omega_2} = \rho_2 g x_2^4 + \rho_2 g x_2 - 2\rho_2 g. \quad (9.112)$$

RESULTS

Figures 9.9 and 9.10 show the log-log plot of the local error in operator B as the mesh is refined at the interface and away from the interface, respectively. The error is calculated in a row near to and far way from the interface at $x_1 = 0.5$. Figures 9.11 and 9.12 show the log-log graph of the local error for operator DG^f as the mesh is refined, at the interface and away from the interface, respectively. A reference line has been drawn in the plots to compare the expected analytical order of accuracy and the numerically obtained order. It can be clearly seen that the numerical order of the accuracy concurs with the analytically calculated order.

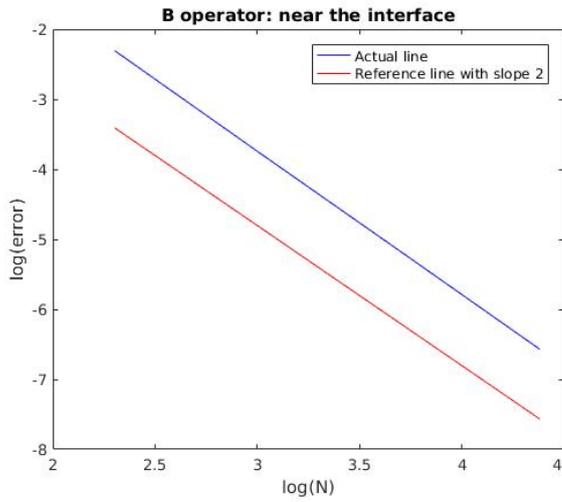


Figure 9.9: Order of the local truncation error in operator B near the interface

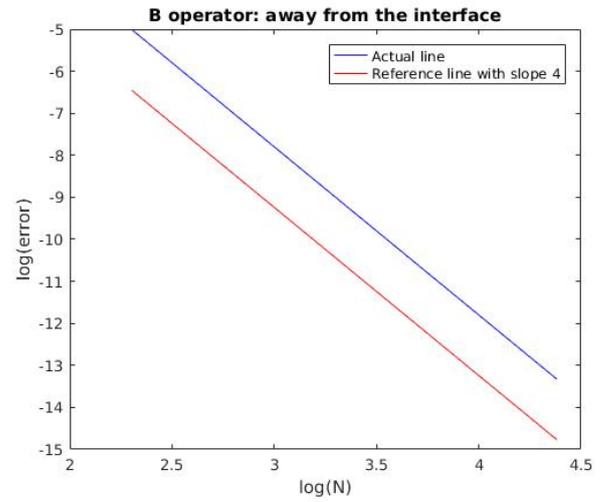


Figure 9.10: Order of the local truncation error in operator B away from the interface

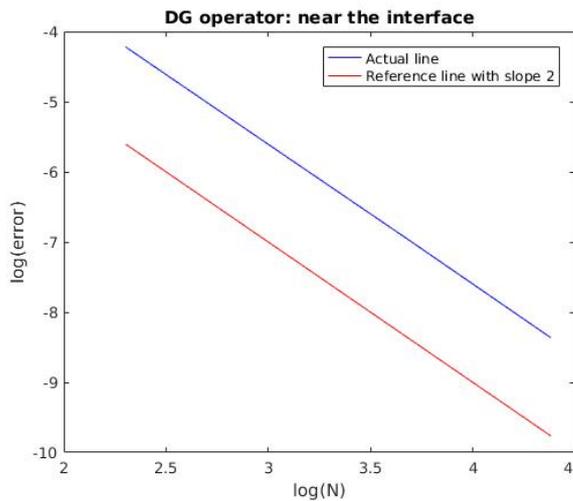


Figure 9.11: Order of the local truncation error in operator DG^f near the interface

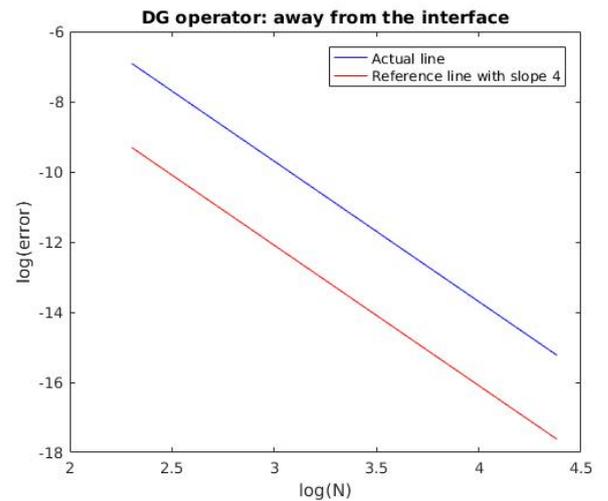


Figure 9.12: Order of the local truncation error in operator DG^f away from the interface

IMPACT OF THE ORDER OF CONVERGENCE

The B operator arises from the pressure velocity coupling formulated on a collocated grid to suppress the pressure-velocity checkerboard modes.

As explained in Section 4.2, the predicted face velocity using the approach of Rhie and Chow pressure velocity coupling is defined as

$$u_{\alpha}^{f*} = (1 - \omega) \left(u_{\alpha}^{fk} - \tilde{u}_{\alpha^c k} \right) + \tilde{u}_{\alpha}^{c*} + \omega \dot{B} p^l. \quad (9.113)$$

Similarly, the corrected face velocity is defined as

$$u_{\alpha}^{f l+1} = (1 - \omega) \left(u_{\alpha}^{fk} - \tilde{u}_{\alpha^c k} \right) + \tilde{u}_{\alpha^c l+1} + \omega \dot{B} p^{l+1}. \quad (9.114)$$

Further, the final continuity equation incorporating the pressure-velocity coupling is given by

$$DG^f \delta p = Du_f^*. \quad (9.115)$$

It can be seen that the the velocity at the face is calculated by linearly interpolating the cell center velocities, and correcting for pressure gradients at the face. The linear interpolation employed is second order accurate, and as seen Section 9.4.2 the DG^f and the B operator is fourth order accurate away from the interface and second order accurate near the interface. Hence, it can be inferred that the constructed operators do not lower the order of accuracy of the discrete system.

9.4.3. DISCUSSION

A general balanced-force formulation for the reconstruction of the flow variables was proposed. An approach to construct the operators, to incorporate the pressure reconstruction consistent with the interface jump conditions was presented. The formulation was implemented for quiescent flow, where the interface is aligned and offset to the control volume faces and an arbitrarily inclined interface. Next, the order of accuracy was derived for the operators and numerically verified using the Method of Manufactured Solutions. It can be concluded that the proposed method successfully produces a discrete force balance, employing the face value reconstruction and a discretization consistent with the interface jump conditions.

Along with the density jump also the viscosity jumps across the interface. A reconstruction for velocity is required to ensure that the velocity interface conditions are satisfied, namely the continuity of the velocity and the tangential shear stress. In the following section, the formulation for the velocity reconstruction is presented. Next, the proposed formulation is implemented for stratified channel flow and the results are discussed.

9.5. VELOCITY RECONSTRUCTION

In an incompressible and immiscible flow the velocity has to be continuous at the interface. Furthermore, the total shear stress acting on the interface should be zero, i.e. the shear stress has to be continuous. The dynamics of the flow are characterized by

$$[\mathbf{u}] = 0, \quad (9.116)$$

$$[\hat{t}^T \cdot \mu \nabla \mathbf{u} \cdot \hat{n}] = 0. \quad (9.117)$$

where, \hat{t} and \hat{n} are the tangential and normal vectors to the interface. The discrete diffusion operator J should satisfy equation (9.116) and (9.117) to ensure a force balance between the viscous forces and the pressure.

In a stratified channel flow, the velocity along the channel forms a piecewise parabolic profile. The net convective fluxes are zero and the diffusion balances the pressure gradient in the channel. If the value of the velocity at the interface and the diffusion operator is carefully formulated to account for the interface jump condition, the piecewise parabolic profile can be numerically replicated. The efficacy of the formulation will be tested for a stratified channel flow.

In this section, first the reconstruction of velocity at the face is presented. Next the results of the implementation of the formulation for stratified flow are discussed. Finally, the order of the formulation is evaluated numerically.

9.5.1. RECONSTRUCTION OF THE VELOCITY AT THE INTERFACE

Reconstruction of the velocity at the face is carried out based on the following jump conditions:

1. Continuity of shear stress

$$[\hat{\mathbf{t}}^T \cdot \mu \nabla \mathbf{u} \cdot \hat{\mathbf{n}}] = 0. \quad (9.118)$$

2. Continuity of velocity

$$[\mathbf{u}] = 0. \quad (9.119)$$

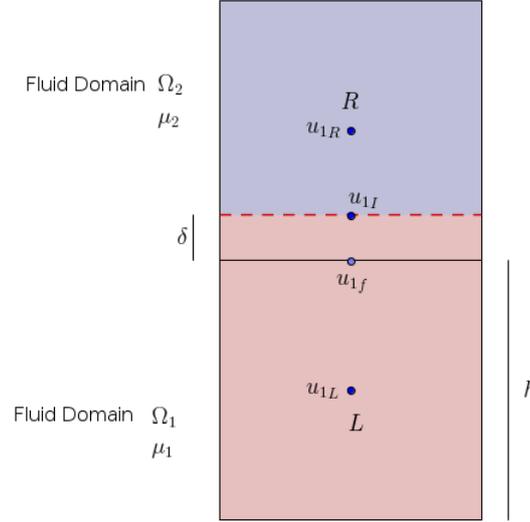


Figure 9.13: Velocity Recinstruction: Example of a offset and aligned interface

Consider Figure 9.13 in which the interface is offset to the common face between control volumes L and R of mesh width h by a distance of δ . Based on the interface conditions, the face value reconstruction employing the general formulation is done:

$$u_{1R} = u_{1I}^R + (0.5h - \delta) u_{1,2}^R + \frac{(0.5h - \delta)^2}{2!} u_{1,22}^R + \frac{(0.5h - \delta)^3}{3!} u_{1,222}^R + \dots \quad (9.120)$$

$$u_{1L} = u_{1I}^L - (0.5h + \delta) u_{1,2}^L + \frac{(0.5h + \delta)^2}{2!} u_{1,22}^L + \frac{(0.5h + \delta)^3}{3!} u_{1,222}^L + \dots \quad (9.121)$$

Multiplying equation (9.120) by μ_1 and dividing by $0.5h - \delta$ and similarly multiplying equation (9.121) by μ_2 and dividing by $0.5h + \delta$, gives

$$\frac{\mu_2 u_{1R}}{0.5h - \delta} = \frac{\mu_2 u_{1I}}{0.5h - \delta} + \mu_2 u_{1,2}^R + \frac{\mu_2 (0.5h - \delta)}{2} u_{1,22}^R + \dots, \quad (9.122)$$

$$\frac{\mu_1 u_{1L}}{0.5h + \delta} = \frac{\mu_1 u_{1I}}{0.5h + \delta} - \mu_1 u_{1,2}^L + \frac{\mu_1 (0.5h + \delta)}{2} u_{1,22}^L + \dots \quad (9.123)$$

In the following sections the reconstruction and the order of the reconstruction are discussed for a zero and non-zero value of δ .

$\delta = 0$

Substituting $\delta = 0$ in (9.122), (9.123) gives

$$\frac{\mu_2 u_{1R}}{0.5h} = \frac{\mu_2 u_{1I}}{0.5h} + \mu_2 u_{1,2}^R + \frac{\mu_2 (0.5h)}{2} u_{1,22}^R + \dots \quad (9.124)$$

$$\frac{\mu_1 u_{1L}}{0.5h} = \frac{\mu_1 u_{1I}}{0.5h} - \mu_1 u_{1,2}^L + \frac{\mu_1 (0.5h)}{2} u_{1,22}^L + \dots \quad (9.125)$$

It is known that

$$\mu_1 u_{1,22}^L = \mu_2 u_{1,22}^R = p_{,1}. \quad (9.126)$$

Hence (9.124) and (9.125) simplify to

$$\frac{\mu_2 u_{1R}}{0.5h} = \frac{\mu_2 u_{1I}}{0.5h} + \mu_2 u_{1,2}^R + 0.5h p_{,1} + \dots \quad (9.127)$$

$$\frac{\mu_1 u_{1L}}{0.5h} = \frac{\mu_1 u_{1I}}{0.5h} - \mu_1 u_{1,2}^L + 0.5h p_{,1} + \dots \quad (9.128)$$

Equations (9.127) and (9.128) are re-written as

$$\frac{\mu_2 u_{1R}}{0.5h} = \frac{\mu_2 u_{1I}}{0.5h} + \mu_2 u_{1,2}^R + 0.5h \left(\frac{\mu_1}{\mu_1 + \mu_2} \right) p_{,1} + 0.5h \left(\frac{\mu_2}{\mu_1 + \mu_2} \right) p_{,1} + \dots, \quad (9.129)$$

$$\frac{\mu_1 u_{1L}}{0.5h} = \frac{\mu_1 u_{1I}}{0.5h} - \mu_1 u_{1,2}^L + 0.5h \left(\frac{\mu_1}{\mu_1 + \mu_2} \right) p_{,1} + 0.5h \left(\frac{\mu_2}{\mu_1 + \mu_2} \right) p_{,1} + \dots \quad (9.130)$$

The velocity at the interface is calculated based on continuity of the shear stress. The first derivatives can be eliminated from (9.129) and (9.130). As a result the numerical solution will satisfy continuity of shear stress, but the cell center velocities calculated will have an error (i.e. the velocity calculated will not be exact) due to an unaccounted fraction of the second derivative of the velocity (or the gradient in pressure) as shown below. Numerical interface velocity is defined as

$$\frac{\mu_\alpha u_{\alpha I}^h}{0.5h} = \frac{\mu_\alpha u_{\alpha I}}{0.5h} + 0.5 \left(\frac{\mu_\alpha}{\mu_1 + \mu_2} \right) h p_{,1}, \quad (9.131)$$

where α is a component of the velocity.

Equations (9.129), (9.130) can be written as

$$\frac{\mu_2 u_{1R}}{0.5h} = \frac{\mu_2 u_{1I}^h}{0.5h} + \mu_2 u_{1,2}^R \left(+0.5h \left(\frac{\mu_1}{\mu_1 + \mu_2} \right) p_{,1} + \dots \right), \quad (9.132)$$

$$\frac{\mu_1 u_{1L}}{0.5h} = \frac{\mu_1 u_{1I}^h}{0.5h} - \mu_1 u_{1,2}^L \left(+0.5h \left(\frac{\mu_1}{\mu_1 + \mu_2} \right) p_{,1} + \dots \right). \quad (9.133)$$

The expression in the brackets shows the fraction of the pressure gradient (or the second derivative of the velocity) that cannot be accounted for in the face value reconstruction and the higher order terms.

Adding the above equations gives

$$\frac{\mu_2 u_{1R}}{0.5h} + \frac{\mu_1 u_{1L}}{0.5h} = \frac{\mu_1 + \mu_2}{0.5h} u_{1I}^h \quad (9.134)$$

$$u_{1h}^h = \frac{\mu_2 u_{1R} + \mu_1 u_{1L}}{\mu_1 + \mu_2} \quad (9.135)$$

Such a treatment ensures that the interface conditions for the interface velocity, namely continuity of the velocity and of the shear stress are satisfied.

Further, from (9.132) and (9.133) we obtain

$$u_{1R} = u_{1I}^h + \mu_2 u_{1,2}^R + \frac{h^2}{4} \left(\frac{\mu_1}{\mu_1 + \mu_2} \right) u_{1,22}^R + \frac{(0.5h - \delta)^3}{3!} u_{1,222}^R, \quad (9.136)$$

$$u_{1L} = u_{1I}^h + \mu_2 u_{1,2}^L + \frac{h^2}{4} \left(\frac{\mu_1}{\mu_1 + \mu_2} \right) u_{1,22}^L + \frac{(0.5h - \delta)^3}{3!} u_{1,222}^L. \quad (9.137)$$

Numerically, the velocities are calculated as

$$u_{1R} = u_{1I}^h + \mu_2 u_{1,2}^R + \mathcal{O}(h^2), \quad (9.138)$$

$$u_{1L} = u_{1I}^h + \mu_2 u_{1,2}^L + \mathcal{O}(h^2). \quad (9.139)$$

Hence, it can be seen that the calculated velocity is second order accurate.

NON-ZERO δ

The velocity at the interface can be derived in a similar fashion as follows.

Adding equations (9.122) and (9.123) and using equation (9.118) gives

$$\frac{\mu_2 u_{1R}}{0.5h - \delta} + \frac{\mu_1 u_{1L}}{0.5h + \delta} = \left(\frac{\mu_2}{0.5h - \delta} + \frac{\mu_1}{0.5h + \delta} \right) u_{1I} + \left(\frac{\mu_2(0.5h - \delta)}{2} + \frac{\mu_1(0.5h + \delta)}{2} \right) u_{1,22}. \quad (9.140)$$

Hence

$$u_{1I} = \frac{\frac{\mu_2 u_{1R}}{0.5h - \delta} + \frac{\mu_1 u_{1L}}{0.5h + \delta}}{\left(\frac{\mu_2}{0.5h - \delta} + \frac{\mu_1}{0.5h + \delta} \right)} + \mathcal{O}(h^2) \quad (9.141)$$

As explained in section 9.5.1, the calculated velocity is second order accurate, and no approximations are made when calculating the first derivative.

9.5.2. RECONSTRUCTION OF VELOCITY AT THE FACE

The value of the velocity at the face is calculated by using a one-sided approximation based on the value of the velocity at the interface. Considering Figure 9.13, the value at the face u_{1f} is calculated by a one sided approximation at the face as follows

$$\begin{aligned} u_{1f} &= u_{1I} - \delta u_{1,2}, \\ u_{1f} &= u_{1I} - \delta \frac{u_{1I} - u_{1L}}{0.5h + \delta}. \end{aligned} \quad (9.142)$$

This can be simplified to

$$u_{1f} = \left(\frac{0.5h}{0.5h + \delta} \right) u_{1I} + \left(\frac{\delta}{0.5h + \delta} \right) u_{1L}. \quad (9.143)$$

9.5.3. DISCRETIZATION

The diffusion operator is given by

$$\begin{aligned} Ju_\alpha &= \int \nabla \cdot \mu \nabla u_\alpha d\Omega \\ &= \int \mu \nabla u_\alpha d\mathbf{S}. \end{aligned} \quad (9.144)$$

Discretizing gives

$$Ju_\alpha = h\mu(u_{\alpha,1} + u_{\alpha,2}). \quad (9.145)$$

In this section, the method of discretization that is consistent with the velocity interface jump conditions and which incorporates the velocity reconstruction is presented.

CONTROL VOLUME AWAY FROM THE INTERFACE

<the discretization away from the interface is given by

$$Ju_1 \cdot \hat{e}_2 = h\mu \left(u_{1,2} \Big|_{i,j-\frac{1}{2}}^{i+\frac{1}{2}} \right) = \mu h \left(\frac{u_{1j+1} - u_{1j}}{h} - \frac{u_{1j} - u_{1j-1}}{h} \right) = \mu h \left(\frac{u_{1j+1} - 2u_{1j} + u_{1j-1}}{h} \right) \quad (9.146)$$

CONTROL VOLUME INTERSECTED BY THE INTERFACE

Consider the evaluation of the \hat{e}_2 component of the operator J in control volume R . The contribution of the operator at the face near the interface $\mathbf{i}_R - \frac{1}{2}\hat{e}_2$ is evaluate by a one-sided approximation, while away from the interface, i.e at $\mathbf{i}_R + \frac{1}{2}\hat{e}_2$, the operator is evaluated by a central difference approximation:

$$Ju_{1R} \cdot \hat{e}_2 = h \left[\mu_2 \left(\frac{u_{1j+1} - u_{1j}}{h} \right) - \mu_1 \left(\frac{u_{1f} - u_{1j-1}}{0.5h} \right) \right] \quad (9.147)$$

CELL BELOW THE INTERFACE

Similarly the \hat{e}_2 component of the operator J in control volume L is evaluated as

$$Ju_{1L} \cdot \hat{e}_2 = h \left[\mu_1 \left(\frac{u_{1f} - u_{1j}}{0.5h} - \frac{u_{1j} - u_{1j-1}}{h} \right) \right]. \quad (9.148)$$

DISCUSSION

It can be seen from (9.147) and (9.148) that at face shared by the control volumes L and R the diffusion operator is evaluated identically (and complies with the interface conditions) to ensure to ensure the diffusive flux is uniquely defined.

9.5.4. ANALYTICAL SOLUTION

The analytical solution for the stratified flow is calculated based on the following interface and boundary conditions:

- The continuity of the velocity at he interface

$$u_{1,I}^1 = u_{1,I}^2 \tag{9.149}$$

- No slip for the \hat{e}_1 component of the velocity at the walls

$$\mathbf{u}_w = 0 \tag{9.150}$$

- The continuity of the shear stress at the interface

$$\mu_1 u_{1,2,I}^1 = \mu_2 u_{1,2,I}^2 \tag{9.151}$$

Employing equations (9.149)-(9.151), the analytical solution is given by

$$u_1^1 = -\frac{p_{,1}}{2\mu_1} x_2^2 + 0.25 p_{,1} \frac{3\mu_1 + \mu_2}{\mu_1(\mu_1 + \mu_2)} x_2 \tag{9.152}$$

$$u_1^2 = -\frac{p_{,1}}{2\mu_2} x_2^2 + 0.25 p_{,1} \frac{3\mu_1 + \mu_2}{\mu_2(\mu_1 + \mu_2)} x_2 + 0.25 p_{,1} \frac{\mu_1 - \mu_2}{\mu_2(\mu_1 + \mu_2)} \tag{9.153}$$

9.5.5. RESULTS

NUMERICAL SOLUTION

The diffusive operator was implemented. Figure 9.14 shows the numerical and analytical results for the velocity along the channel for $\mu_1 = 120$, $\mu_2 = 1$. It can clearly be seen that the numerical solution reproduces the piecewise parabolic profile, as expected analytically.

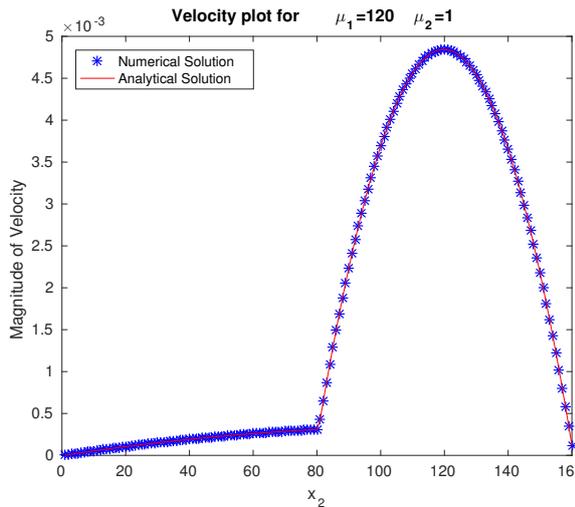


Figure 9.14: Stratified flow: Plot of numerical and analytical solution of velocity

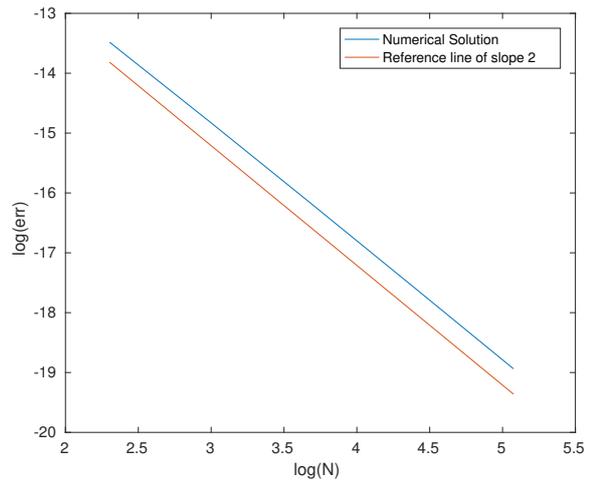


Figure 9.15: Order of velocity reconstruction near the interface

ORDER OF OPERATOR J

Figure 9.15 shows the log–log plot of the absolute of global error of the velocity as a function of the grid size for $\mu_1 = 120$ and $\mu_1 = 1$. As the log–log plot has a slope of 2 it can be concluded that the operator J is second order accurate.

10

INTEGRATED SOLVER

A balanced-force method based on the interface jump conditions, which involves the reconstruction of values near the interface and a consistent construction of operators, was presented in Chapters 8 and 9.

It is sufficient for a stationary discrete formulation to comply with the differential equation and the interface jump conditions. However, to apply this approach to solve unsteady flow problems the influence of the reformulated operators on the stability properties of the system has to be investigated. The properties of the individual operators and their behavior when they are embedded in the complete solver algorithm are analyzed. Results are shown for dynamic flow test cases and compared with the numerical results obtained with OpenFOAM.

10.1. NAVIER-STOKES SOLVER

The complete solver consists of solving for the velocities and the pressure iteratively. To this end, the continuity equation and the conservation of momentum equations in both coordinate directions and the advection of level set are solved. The fluids are considered to be incompressible and immiscible.

Miller and Schmidt [13] proposed an iterative, segregated solver for stationary single phase flow as explained in section 4.2. Based on the Miller and Schmidt method an iterative method for transient multiphase flows is proposed as follows.

The solution algorithm consists of the following sequence of steps:

start of time loop

(iteration level n)

start of outer loop

(iteration level k)

1. Construct the diffusion operator J
2. Construct operators DG^f , B^c , G^c , G^f and the convective operator N

start of inner loop

(iteration level l)

1. The momentum equations are solved to calculate the predicted cell center velocities $\mathbf{u}_1^{c*}, \mathbf{u}_2^{c*}$. The convective term in the right hand side is calculated using outer loop face and cell center velocities $(\mathbf{u}_1^{c k}, \mathbf{u}_2^{c k}, \mathbf{u}_1^{f k}, \mathbf{u}_2^{f k})$.

$$\mathbf{u}^{c**} = J\mathbf{u}^{c*} + N\mathbf{u}^k - G^c \mathbf{p} \quad (10.1)$$

The predicted cell center velocity is under-relaxed, by a under-relaxation factor ω , with respect to the cell center velocity calculated at the last iterate level of the outer loop.

$$\mathbf{u}^{c*} = \omega\mathbf{u}^{c**} + (1 - \omega)\mathbf{u}^k \quad (10.2)$$

2. The predicted face velocities are calculated based on the predicted cell center velocities $\mathbf{u}_1^{f*}, \mathbf{u}_2^{f*}$.

$$\mathbf{u}^{f*} = (1 - \omega) (\mathbf{u}^{fk} - \tilde{\mathbf{u}}^{ck}) + \tilde{\mathbf{u}}^{c*} + \omega \dot{B} \mathbf{p}^l \quad (10.3)$$

3. A correction in pressure is calculated by imposing the solenoidality condition on the corrected face velocities δp . The corrected face velocity is defined as

$$\mathbf{u}^{fl+1} = (1 - \omega) (\mathbf{u}^{fk} - \tilde{\mathbf{u}}^{ck}) + \tilde{\mathbf{u}}^{ck+1} + \omega \dot{B} \mathbf{p}^{l+1}. \quad (10.4)$$

Imposing the solenoidality condition and simplifying, gives

$$DG^f \delta \mathbf{p} = D\mathbf{u}^{f*} \quad (10.5)$$

4. The cell center velocities are corrected based on the calculated correction in the pressure: u_1^{ck+1}, u_2^{ck+1}

$$u_\alpha^{cl+1} = u_\alpha^{c*} - \frac{\omega}{1 - \omega} \frac{\Delta t}{\rho} \delta p_{,\alpha c} \quad (10.6)$$

5. The face velocities are corrected based on the calculated correction in the pressure u_1^{fk+1}, u_1^{fk+1}

$$u_\alpha^{fl+1} = u_\alpha^{f*} - \frac{\omega}{1 - \omega} \frac{\Delta t}{\rho} \delta p_{,\alpha f} \quad (10.7)$$

6. The pressure is updated

$$\mathbf{p}^{l+1} = \mathbf{p}^l + \delta \mathbf{p} \quad (10.8)$$

end of inner loop

The velocities in the outer loop are updated to the values obtained in the latest iteration level of the inner loop.

end of outer loop

The velocities at the new time level are updated to the values obtained in the latest iteration level of the outer loop.

end of time loop

The presented method for instationary flows and the approach proposed by Miller and Schmidt[13] differ in the following aspects

1. The approach presented by Miller and Schmidt has two loops: an inner and an outer loop. The predicted cell center velocities are calculated using the linearized convection operator, and the convection and the diffusion operator are treated implicitly.

The above proposed method treats diffusion implicitly and calculates the convective term in the outer loop. The outer loop is iterated until it converges, hence the treatment of the convective operator is implicit.

2. Miller and Schmidt calculate a one step correction in the pressure in the inner loop, i.e the correction in the pressure is calculated with respect to the pressure in the outer loop. In the method presented, the correction in the pressure is calculated incrementally until the inner loop and the outer loop converge, i.e. the correction in the pressure is calculated with respect to the pressure at the last iteration level in the inner loop.

The complete solver, incorporating the balanced-force framework, was evaluated for the cases where the computational domain is closed and for flow in a rectangular channel. A fixed number of inner and outer iterations was used, determined by ensuring that the inner and the outer loops have converged before stepping in time. The formulation of the discretisation differs only through the different imposed boundary and initial conditions. The results of the numerical simulation are discussed in the upcoming section.

10.2. CASE: CLOSED BOX

In this case, the flow in a closed box or tank is numerically modeled. The fluids are allowed to slip along the boundaries of the domain, i.e. the tangential stress on the boundary is zero. Thereto the imposed boundary conditions are as follows (graphically represented in Figure 10.1):

- **Left and Right Boundary**
 - Pressure- Homogeneous Neumann
 - Velocity in \hat{e}_1 - Homogeneous Dirichlet
 - Velocity in \hat{e}_2 - Homogeneous Neumann
- **Top and Bottom Boundary**
 - Pressure- Non-homogeneous Neumann
 - Velocity in \hat{e}_1 - Homogeneous Neumann
 - Velocity in \hat{e}_2 - Homogeneous Dirichlet

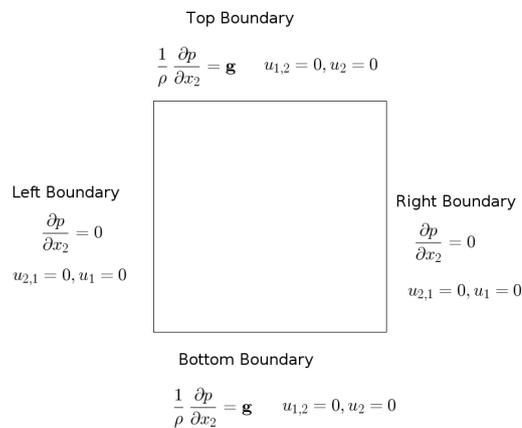


Figure 10.1: Closed box case: Boundary conditions

HANDLING OF PRESSURE BOUNDARY CONDITIONS

The evaluation of the pressure at a row/column of control volumes outside the computational domain, i.e the ghost nodes, is required in the formulation of operator B^c . The possible approaches for the this treatment are:

- Employ the boundary conditions to determine the value at the ghost node.
In this approach, the boundary condition is utilized to calculate the values of the pressure outside of the computational domain. The calculation is based over a distance of one mesh width.
- Extrapolate the value of the pressure at the ghost nodes, based on the values inside the computational domain.
Wesseling [22] and Peric *et al.* [23] recommend extrapolation to calculate the value of pressure at the ghost node, albeit with giving a numerical justification. The extrapolation is performed over a distance of two mesh widths.

For a single phase flow the operators have to satisfy the relation $D^T = -G$. Extrapolation of the pressure at the boundaries violates this equality, whereas if the boundary conditions are employed the equality is satisfied (only for single phase flow). Furthermore, it leads to a higher numerical accuracy (one mesh width versus two). Hence the final evaluation of pressure was performed by utilizing the boundary condition.

DYNAMIC FLOW CASES

The fully closed problem numerical setup was evaluated for the following cases:

- Quiescent flow
- A viscous standing wave in a tank
- Dam break case

The results of the dynamic cases are presented in the following subsections.

10.2.1. QUIESCENT FLOW

The balanced-force formulation of the operators B and DG^f was evaluated for a quiescent flow for the case in which the interface coincides with the cell faces, the interface is aligned but offset with the cell faces and in which the interface is neither coinciding nor aligned with the cell faces; see Section 8.2.3, 9.2.4 and Section 9.3, respectively. The numerical experiments show that the operators produce a discrete force balance upto machine precision.

In this section, the complete frame work is evaluated, where the discrete system is numerically solved for both the velocities and the pressure for quiescent flow. First, the convergence characteristics of the solver are studied for an unperturbed initialization. Next, the case is studied, where the pressure is perturbed by a cosine wave to investigate the effect of modifying the formulation of the pressure gradient and the effect of the operator B on the stability of the system.

The quiescent case is investigated in the following section for the case where the interface is offset and aligned with the faces of the control volume. The error plots of the velocities are studied to evaluate the capability for the complete numerical framework to produce a discrete force balance and suppress numerical artefacts.

The quiescent flow is evaluated for the following computational and fluid properties:

Computational properties

- Size of the domain: 1×1
- Mesh width (h/L): 0.2 (Number of control volumes: 5×5)
- Time step size (Δt): 0.1
- Location of the interface: interface offset by $0.28h$
- Number of inner loop iterations per outer loop: 30
- Number of outer loop iterations per time step: 15

Fluid properties

- Density ratio: $\frac{\rho_1}{\rho_2} = 0$
- Viscosity ratio: $\frac{\mu_1}{\mu_2} = 100$

NO PERTURBATION

Figures 10.2 and 10.3 show the plot of the L^∞ norm of the error for the face velocities against the number of inner loop iterations. The order of the error has a magnitude of approximately 10^{-31} . Furthermore, figure 10.4 and 10.5 show the plot of the error in the cell center velocities against the number of inner loop iterations. The error in the \hat{e}_1 component of the velocity has a magnitude 10^{-31} , whereas the error in the component \hat{e}_2 of the cell center velocity has that of magnitude 10^{-16} . It can be seen that the errors are of the magnitude of machine precision. The difference in the order of the error in between u_2^f and other cell center and face velocities, is attributed to the fact that the face velocities are made divergence free in the pressure correction equation, further the projection of the pressure gradient in the direction \hat{e}_1 is zero, whereas in the direction \hat{e}_2 it is equal to the density scaled gravitational acceleration, hence providing more opportunity for the accumulation of the round off errors.

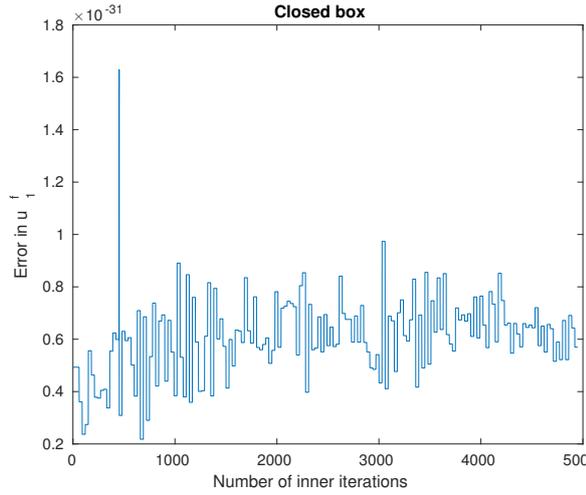


Figure 10.2: Offset and aligned interface: Plot of L^∞ norm of error in u_1^f

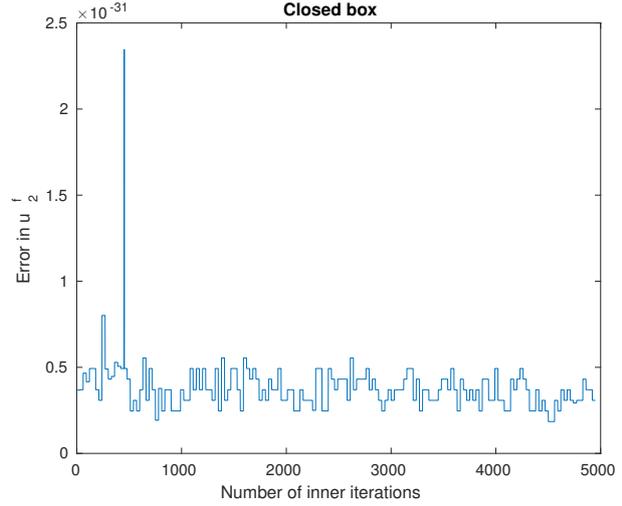


Figure 10.3: Offset and aligned interface: Plot of L^∞ norm of error in u_2^f

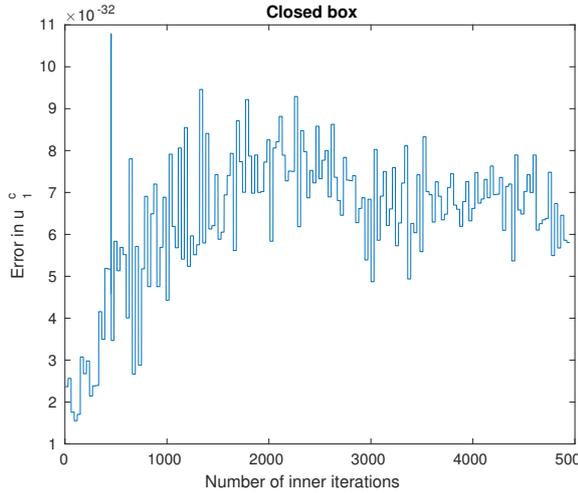


Figure 10.4: Offset and aligned interface: Plot of L^∞ norm of error in u_1^c

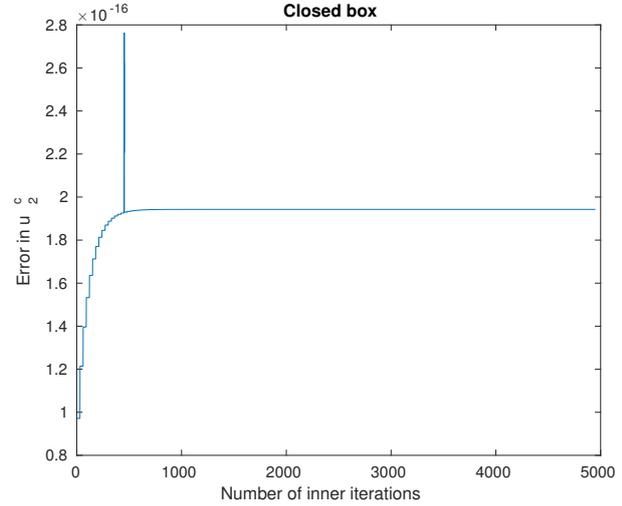


Figure 10.5: Offset and aligned interface: Plot of L^∞ norm of error in u_2^c

PERTURBED INITIALIZATION

The pressure is initialized with a perturbation of a cosine wave of magnitude 10^{-12} in the direction \hat{e}_2 (i.e. aligned with gravity) with respect to the hydrostatic pressure, as shown in Figure 10.6. The expression for the perturbation in the pressure is given by

$$\Delta_p = a \cos(2\pi x_2) - a \quad (10.9)$$

The purpose of this investigation is to determine the behaviour of the operators when a flow variable is perturbed and to verify the effectiveness of the pressure velocity coupling. Figure 10.7 shows the plot of the maximum absolute value of the error in pressure as a function of the inner loop iterations. It is clearly seen, that the initialized perturbation gradually damps out, and in a few inner loop iterations the oscillation in the pressure has completely disappeared. Figures 10.8, 10.9 and Figures 10.10 and 10.9 show the plots of the L^∞ norm of the error in the face velocities and the cell center velocities as a function of the number of inner loop iterations, respectively. As can be seen in the unperturbed case, the error in the velocities is maintained upto the machine precision.

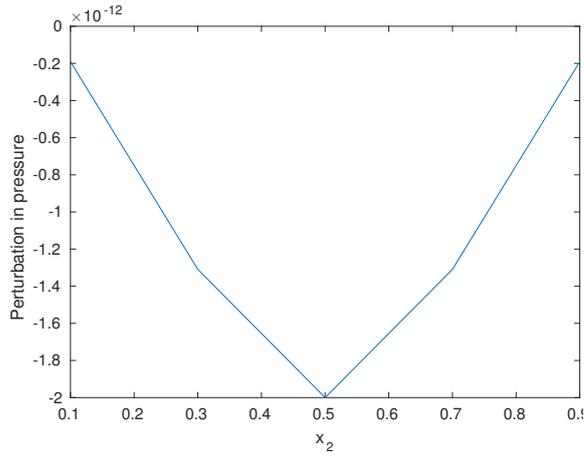


Figure 10.6: Offset and aligned interface: Plot of initialized perturbation in pressure against x_2

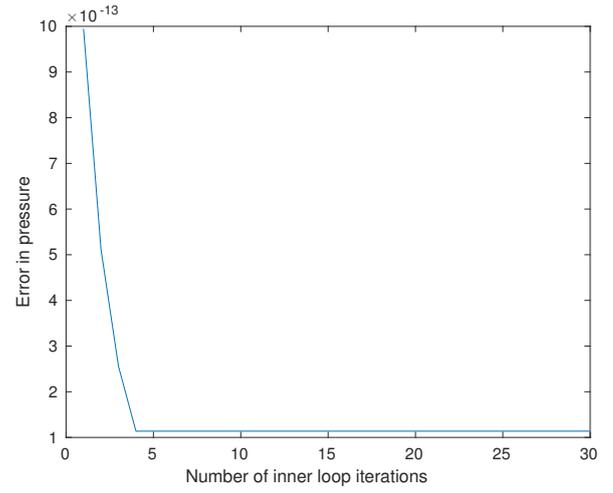


Figure 10.7: Offset and aligned interface: Plot of L^∞ norm of error in pressure

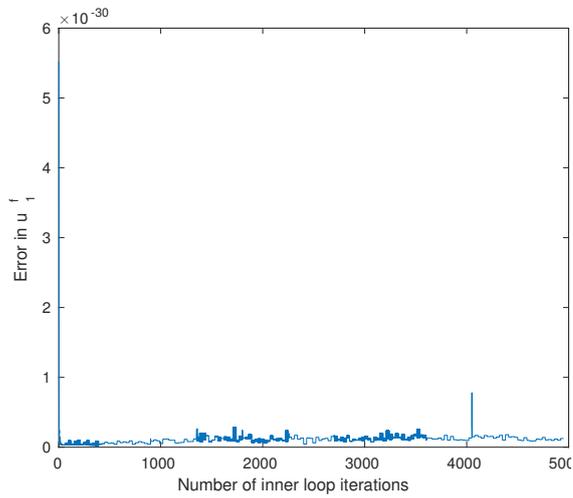


Figure 10.8: Offset and aligned interface perturbed case: Plot of L^∞ norm of the error in u_1^f

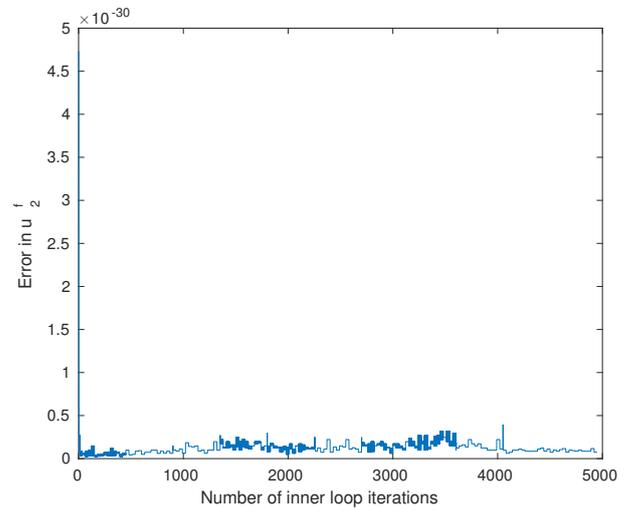


Figure 10.9: Offset and aligned interface perturbed case: Plot of L^∞ norm of the error in u_2^f

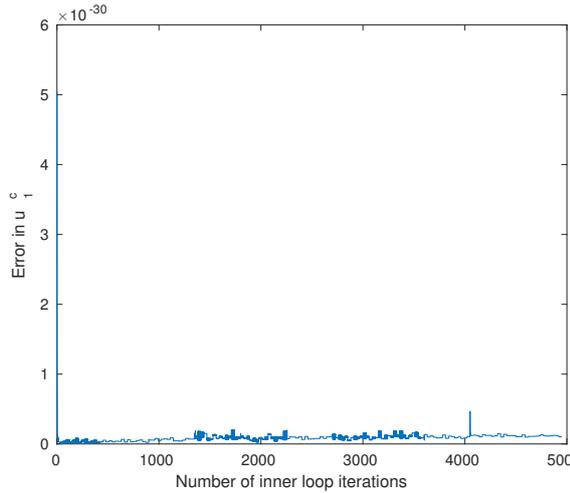


Figure 10.10: Offset and aligned interface perturbed case: Plot of L^∞ norm of the error in u_1^c

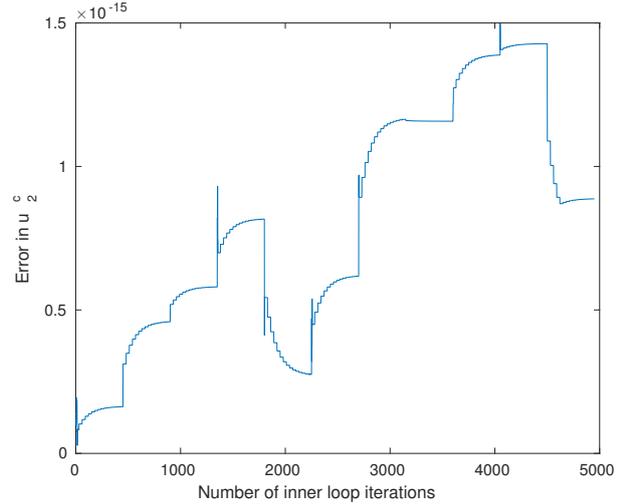


Figure 10.11: Offset and aligned interface perturbed case: Plot of L^∞ norm of the error in u_2^c

INFERENCES

The following inferences can be drawn from the results presented in this section:

- The proposed formulation achieves a discrete force balance and avoids the occurrence of spurious velocities at the interface for the case where the interface does not coincide with the cell faces.
- The balanced-force formulation of the pressure-velocity coupling successfully removes the introduced numerical oscillations and maintains the solution upto machine accuracy.
- The iterative solver constructed with the force balanced operators is successful in correcting the pressure and velocity field to produce a divergence free flow.

10.2.2. VISCOUS WAVE

Greaves *et al.* [24] and Wang *et al.* [25] numerically modeled a viscous standing wave in a tank using adaptive quad tree grids. A similar viscous wave is numerically modeled to investigate the behavior of the solver in a dynamic case.

A perturbed wave of small amplitude is initialized as follows:

$$\eta = a \cos(2\pi x b), \quad (10.10)$$

where $b = 2h$ is the length of the tank, h is the mean water depth and $a = 0.002h$ is the amplitude of the wave. The length of the tank was assumed to be $b = 1$. Figure 10.12 shows the contour plot of the level set field. The different colours indicate the two fluids in the domain. The damping of the wave over time is modeled for a Reynolds number defined as $Re = \frac{h\sqrt{gh}}{\nu}$. The elevation of the wave is studied to determine the damping of the wave at different Reynolds numbers. The larger the Reynold number, the larger the inertial forces acting on the fluids and the longer it takes for the wave to damp. In the following subsection the results for different Reynolds numbers are discussed and compared with the OpenFOAM solutions.

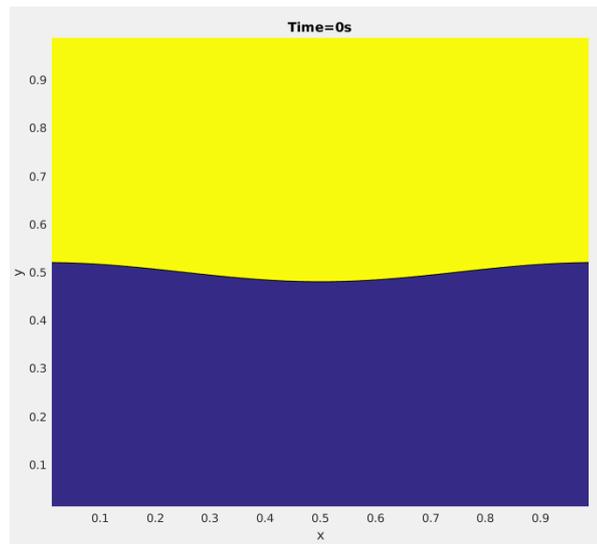


Figure 10.12: Closed box: Initialized viscous standing wave

$Re = 2$

Mesh details

Figure 10.13 shows the elevation of the wave at $x_1 = 0.5$ as a function of time for the balanced-force formulation at different mesh sizes and the OpenFOAM solution. OpenFOAM employs a volume of fluid based solver where the fluid properties and the interface are diffused over the control volumes. As a result, OpenFOAM does not recognize a perturbed interface for the initialization (10.10) for a coarse mesh (such as 10×10), necessitating a much finer mesh.

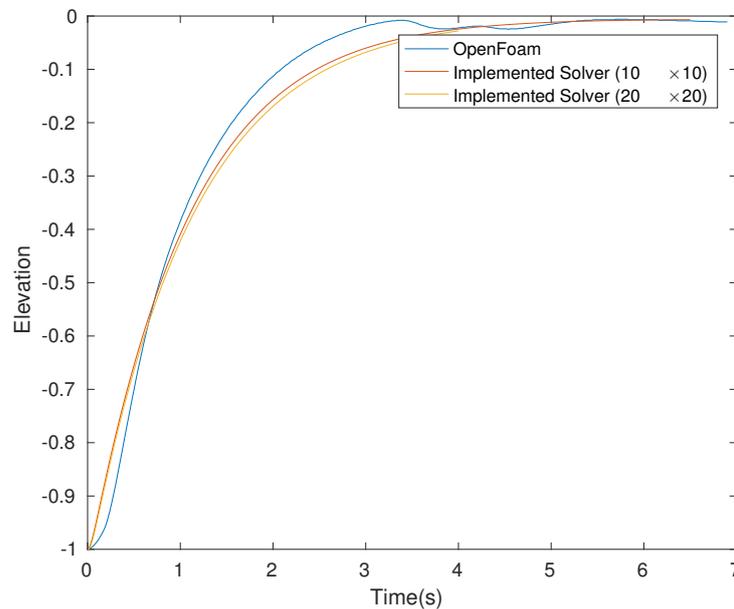


Figure 10.13: Viscous standing wave: Elevation plot for $Re = 2$

A similar damping of the wave is seen in the numerical solution of OpenFOAM and the implemented force balanced solver. No significant differences are seen when the mesh is refined.

Quiver Plots

Figure 10.14 and 10.15 show the quiver plots of the velocity for the implemented solver for the solution obtained with a mesh with $h/b = 0.1$ and OpenFOAM, respectively. OpenFOAM produces spurious velocities at

the boundaries and at the interface. Furthermore, the velocity profile is not symmetric and perturbations in the velocity are seen near the center of the vortices. On the other hand, the new solver produces a symmetric and smooth profile, indicating continuity of the velocity across the interface.

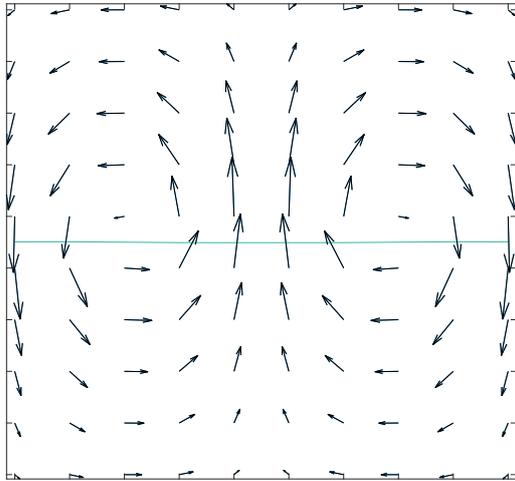


Figure 10.14: Viscous standing wave: Quiver plot at $t = 2.5s$ for the Implemented Solver

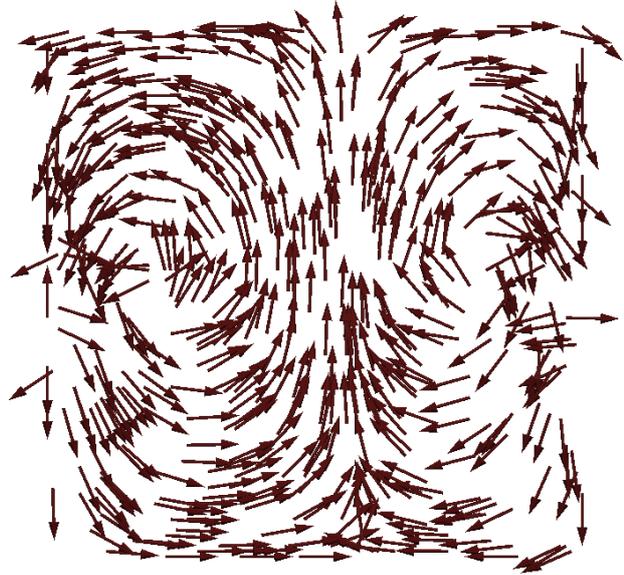


Figure 10.15: Viscous standing wave: Quiver plot at $t = 2.5s$ for OpenFoam

In the following subsections the flow characteristics are evaluated by comparing the numerical solution of OpenFoam and the solution obtained from the implemented solver for a mesh of 10×10 .

Velocity

Figure 10.16 shows the magnitude of velocity along a vertical line at $x_1 = 0.5$. The velocity is expected to vary smoothly along this vertical line with the highest value expected at $x_2 = 0.5$ near the interface. An indication of spurious velocities can be seen near the interface ($x_2 = 0.5$) in the solution of OpenFOAM, whereas the implemented solver leads to a smooth (i.e. no unphysical velocity spikes) profile for the velocity.

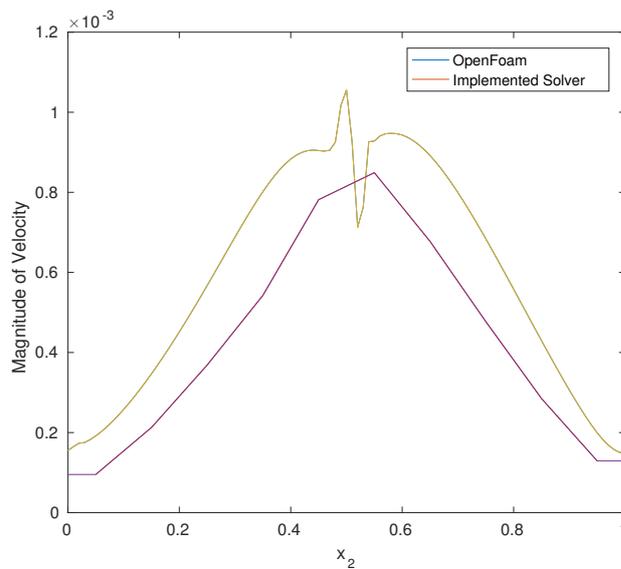


Figure 10.16: Viscous standing wave: Comparison of velocity at $x_1 = 0.5$

Figures 10.17 and 10.18 show the magnitude of velocity along a horizontal line at $x_2 = 0.5$. As the horizontal

line at $x_2 = 0.5$ is traversed, the magnitude of the velocity should decrease as one reaches the center of the vortices and it should increase away from it. Such a behavior is seen in the solution obtained from the formulated solver, while OpenFoam produces spurious velocities of a much higher magnitude at the boundaries of the closed box.

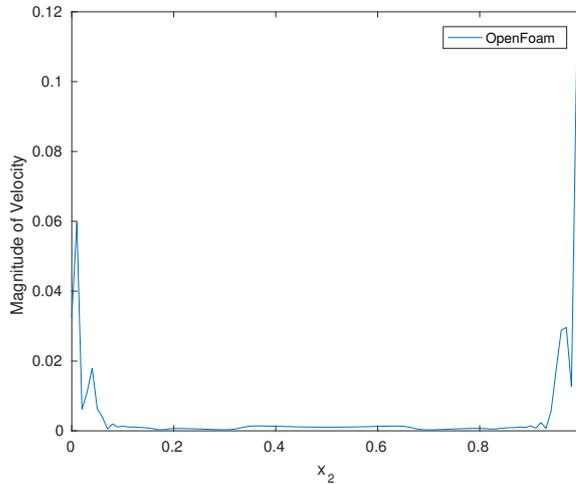


Figure 10.17: Viscous standing wave: Velocity plot at $x_2 = 0.5$ for the Implemented Solver

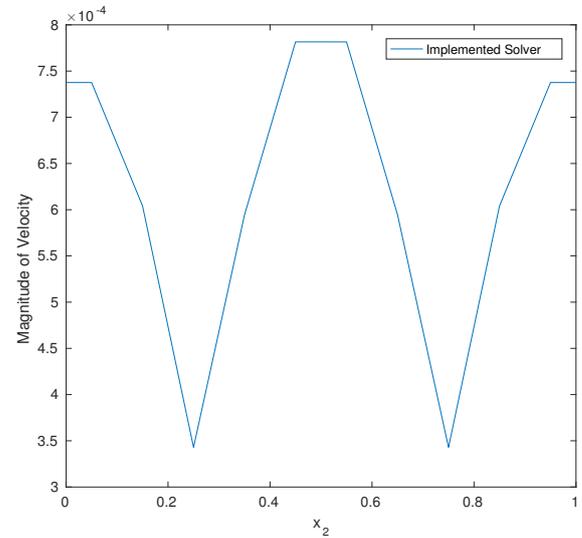


Figure 10.18: Viscous standing wave: Velocity plot at $x_2 = 0.5$ for OpenFoam

Implemented solver: Mesh refinement

Figures 10.19 and 10.20 show velocity along the lines $x_1 = 0.5$ and $x_2 = 0.5$, respectively for the mesh of 10×10 and 20×20 . No differences are seen in the dynamics of the solution. The velocity profiles show a smooth distribution with no indication of spurious contributions.

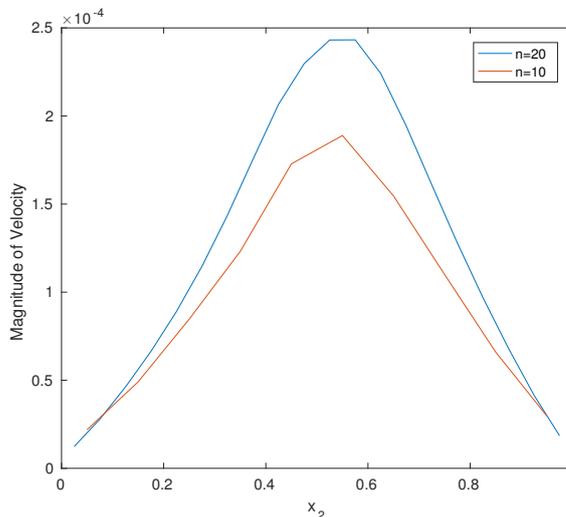


Figure 10.19: Mesh refinement: Velocity plot along the vertical line at $x_1 = 0.5$

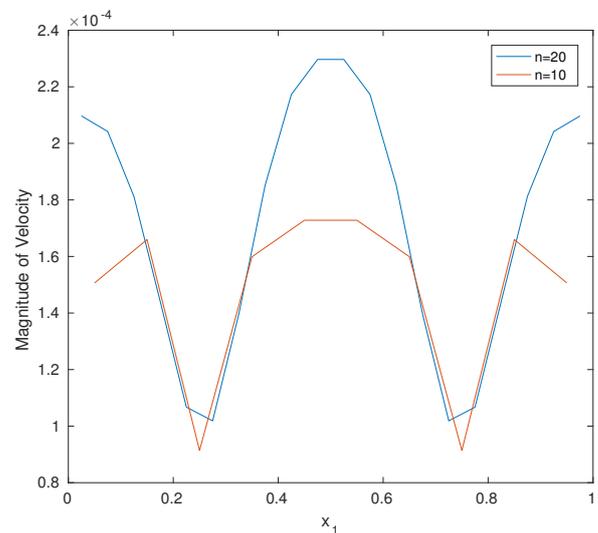


Figure 10.20: Mesh refinement: Velocity plot along the vertical line at $x_2 = 0.5$

HIGHER REYNOLDS NUMBER: $Re = 20$ AND $Re = 200$

Figure 10.22 and 10.22 show the elevation for Reynolds number $Re = 20$ and $Re = 200$, respectively. It can be seen that the formulated solver and the numerical solution of OpenFOAM show a similar trend. A difference

is seen in the elevation plots when the mesh is refined for the formulated solver. This is attributed to the fact that the flow at a higher Reynolds number is much more violent, and hence needs a much finer mesh to accurately model the flow.

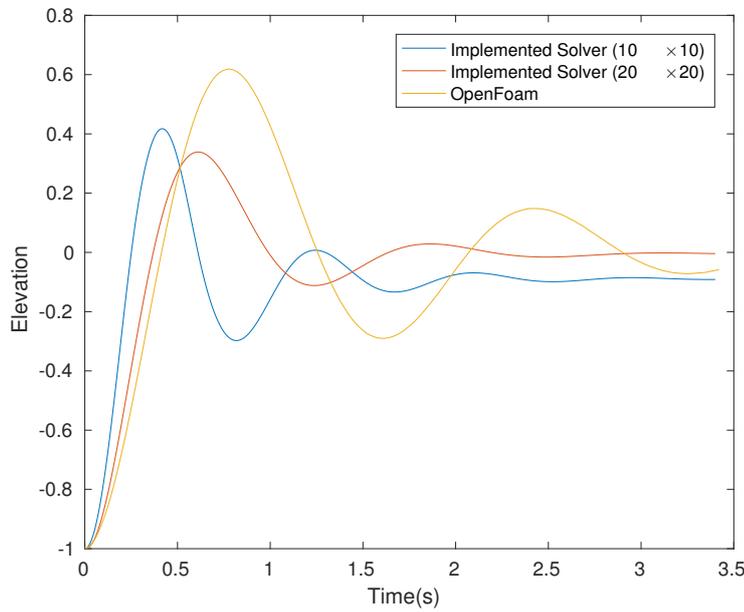


Figure 10.21: Viscous standing wave: Elevation plot at $Re = 20$

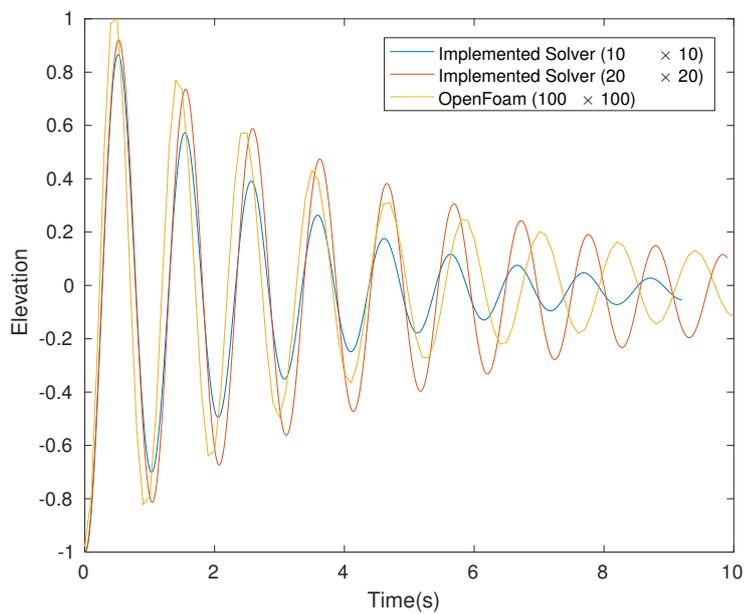


Figure 10.22: Viscous standing wave: Elevation plot at $Re = 200$

A similar occurrence of spurious velocities is observed in the numerical solution of OpenFOAM for higher Reynolds numbers. OpenFOAM fails to maintain a smooth symmetric velocity profile, similar to what is seen in figure 10.14 for $Re = 2$. On the contrary, the formulated solver produces a smooth continuous flow profile, without occurrence of spurious velocities. The plots have been omitted for the sake of conciseness and to avoid repetition of information.

10.2.3. DAM BREAK

A dam break case is modeled by initializing an inclined interface. An obstacle of one mesh width by two mesh widths is placed at $x_1/b = 0.2$ as shown in solid red line in Figure 10.23. The computational and fluid properties employed are as follows:

Computational properties

- Mesh width (h/b): 0.05 (20×20)
- Time step size (Δt): 0.01
- Location of the interface: $x_2/b = 0.4$
- Orientation of the interface: slope 0.3

Fluid properties

- Density ratio: $\frac{\rho_1}{\rho_2} = 50$
- Viscosity ratio: $\frac{\mu_1}{\mu_2} = 50$

The flow is allowed to slip at the boundaries and at the walls of the obstacle. As the inclined interface oscillates, a wave is created at the interface due to the presence of the obstacle. The developed wave breaks over the interface and eventually is expected to damp out to a steady state quiescent flow.

Figures 10.23-10.26 shows the quiver plot of the velocity, the location of the interface and the obstacle at $t = 0.1s$, $t = 0.2s$, $t = 0.3s$ and $t = 0.7s$, respectively.

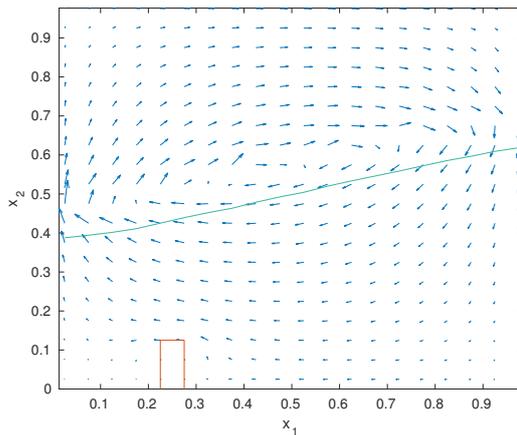


Figure 10.23: Dam Break: Quiver plot of velocity and contour of the level set field at $t=0.1s$

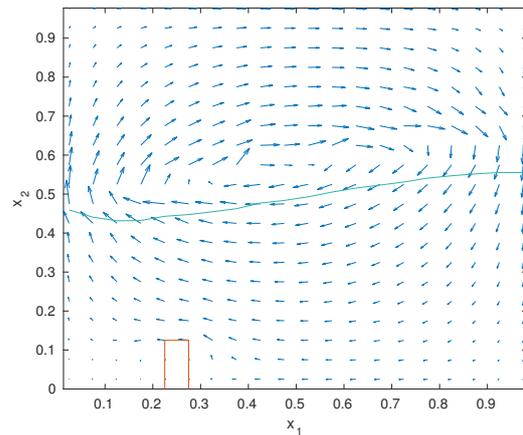


Figure 10.24: Dam Break: Quiver plot of velocity and contour of the level set field at $t=0.2s$

Figure 10.26 shows that a wave is formed at the left boundary. The wave travels to the right as time progresses, as shown in Figures 10.28-10.30.

Figures 10.27-10.30 show the contour of the level set field at $t = 0.3s$, $t = 0.4s$, $t = 0.7s$ and $t = 0.8s$, respectively. It can be seen that the level set field is not smooth. This is attributed to the fact that a coarse mesh is being employed and requires a finer mesh for high inertia flows. As the simulation progresses, when the wave crest breaks against the interface, the edges of the wave are a relatively short distance apart compared to the mesh width. As a large stencil is employed in the interface model, over the course of further time steps it leads to a tear in the level set field. Hence a much smaller mesh width may be required to avoid an incorrect advection of the level set field. Due to limited computational and temporal resources a finer mesh could not be used.

The viscous standing wave and the dam break case successfully illustrate the effectiveness of the balanced-force sharp interface method. The numerical simulations performed and the comparison made between the numerical solution obtained by the implemented solver and OpenFOAM demonstrate that the proposed formulation avoids occurrence of spurious velocities and numerical dispersions at the interface. Furthermore, the pressure-velocity algorithm proposed, eliminates oscillations in the flow fields.

In the next section the analysis and the results of two-phase stratified channel flow are discussed.

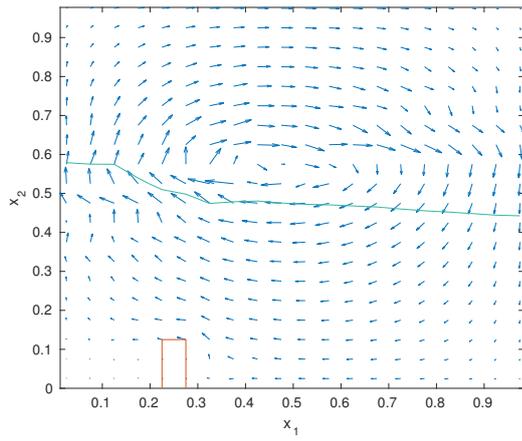


Figure 10.25: Dam Break: Quiver plot of velocity and contour of the level set field at $t=0.3s$

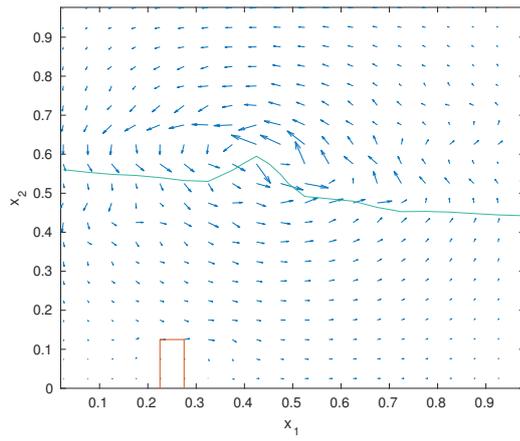


Figure 10.26: Dam Break: Quiver plot of velocity and contour of the level set field at $t=0.7s$

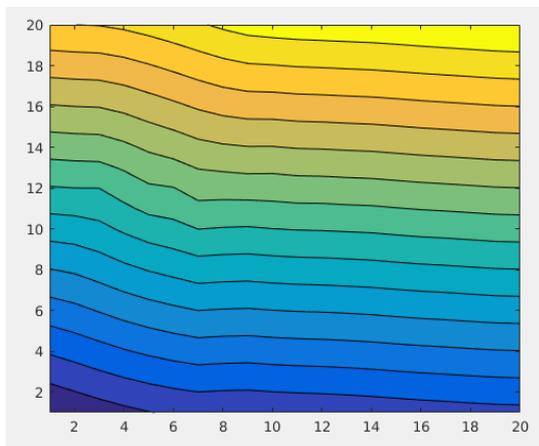


Figure 10.27: Dam Break: Contour of level set field: $t=0.3s$

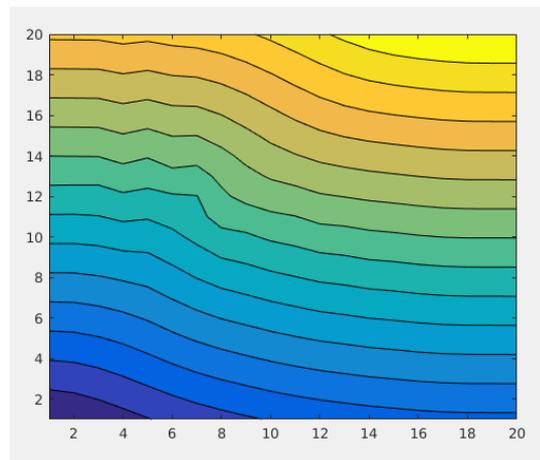


Figure 10.28: Dam Break: Contour of level set field: $t=0.4s$

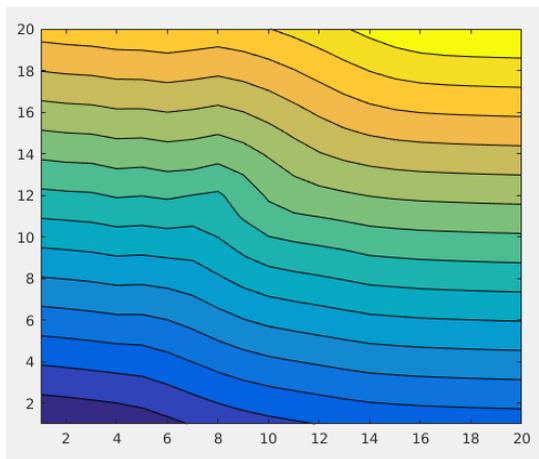


Figure 10.29: Dam Break: Contour of level set field: $t=0.7s$

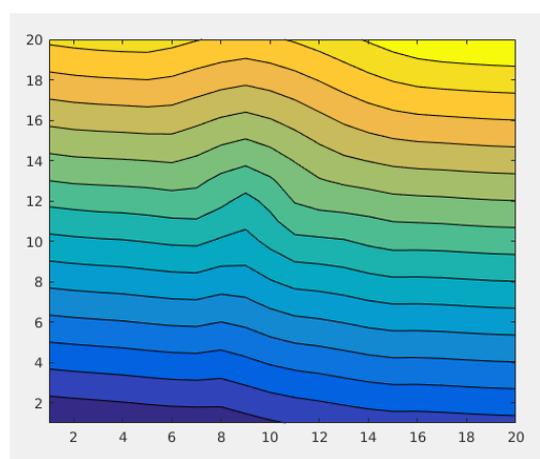


Figure 10.30: Dam Break: Contour of level set field: $t=0.8s$

10.3. STRATIFIED FLOW

Stratified flow consists of multiphase flows with a density variation in the vertical direction, where a lighter fluid lies on top of a denser fluid. In this case, a two-phase stratified channel flow is numerically modeled. Numerically it can be modeled with two sets of boundary conditions:

- Inflow-Outflow
- Periodic boundary conditions

In the following section the results and analysis of the stratified case for the two choices of boundary conditions are discussed.

10.3.1. INFLOW-OUTFLOW BOUNDARY CONDITIONS

The following boundary conditions are imposed:

- **Top and Bottom Boundary**
 - Pressure- Non-homogeneous Neumann
 - Velocity in \hat{e}_1 - Homogeneous Neumann
 - Velocity in \hat{e}_2 - Homogeneous Dirichlet
- **Left and Right Boundary**
 - Pressure
 - ◊ Left- Homogeneous Neumann
 - ◊ Right- Non-homogeneous Dirichlet
 - Velocity in \hat{e}_1 - Homogeneous Dirichlet
 - Velocity in \hat{e}_2 - Homogeneous Neumann

Figure 10.31 graphically shows the imposed boundary conditions.

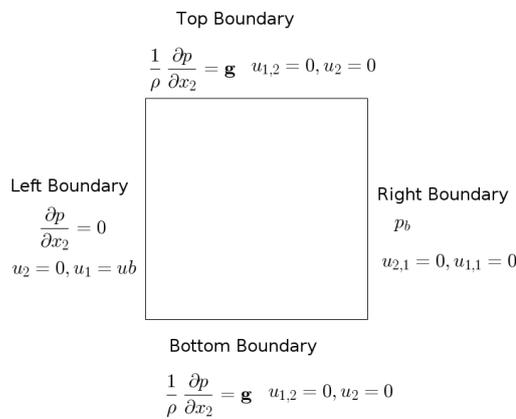


Figure 10.31: Stratified flow: Inflow-Outflow condition

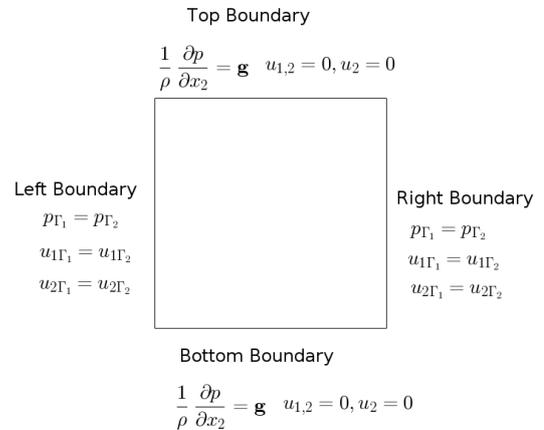


Figure 10.32: Stratified flow: Periodic Boundary condition

With the imposed boundary conditions the tangential stress on the top and bottom is zero. A Dirichlet condition is imposed for the pressure evaluated according to the location of the interface. As a result the operators B and DG become non-singular. As explained in Section 10.2, at the boundaries the pressure at the ghost nodes is calculated by employing the specified boundary condition.

The computational and fluid properties defined in Section 10.2.1 and a uniform velocity of $u_1 = 1$ are utilized.

ANALYSIS OF ITERATIVE SOLVER

Figures 10.33, 10.36 show the L^∞ norm of the error in the face and the cell center velocities as a function of the inner loop iterations.

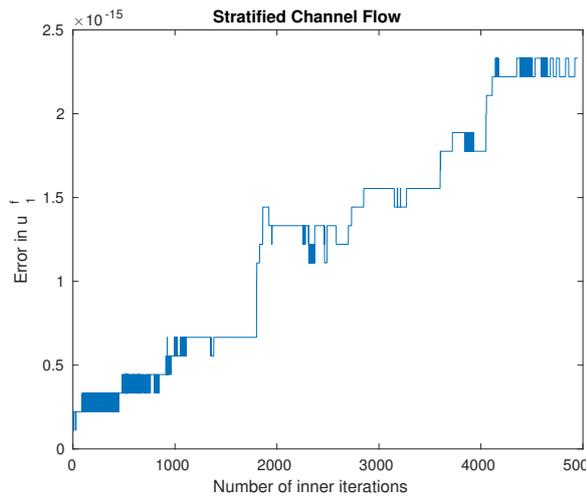


Figure 10.33: Stratified flow: Plot of L^∞ norm of the error in u_f^1

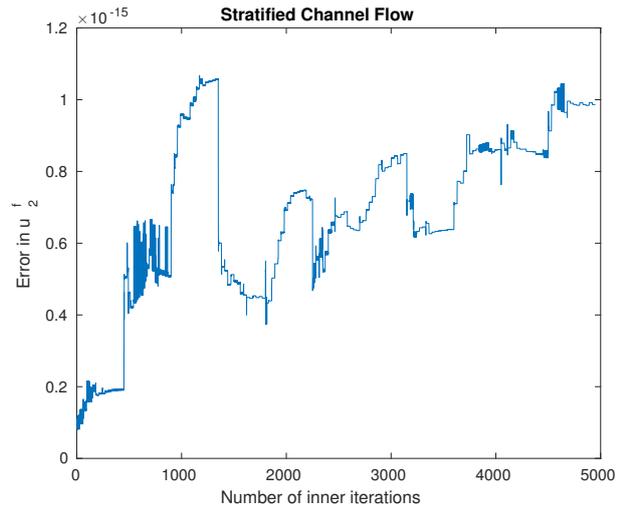


Figure 10.34: Stratified flow: Plot of L^∞ norm of the error in u_f^2

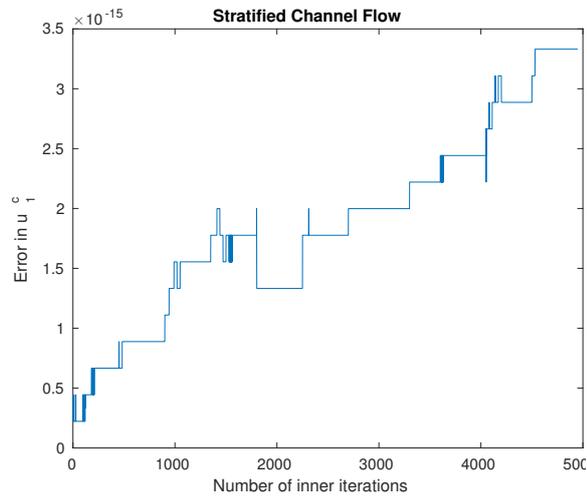


Figure 10.35: Stratified flow: Plot of L^∞ norm of the error in u_c^1

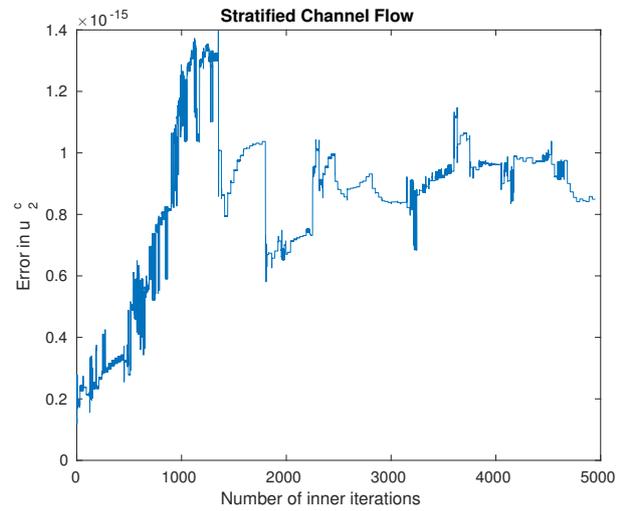


Figure 10.36: Stratified flow: Plot of L^∞ norm of the error in u_c^2

Figures 10.37 to 10.40 show L^∞ norm of the error in the velocities as a function of the number of inner loop iterations for the case, where the pressure is perturbed as explained in Section 10.2.1.

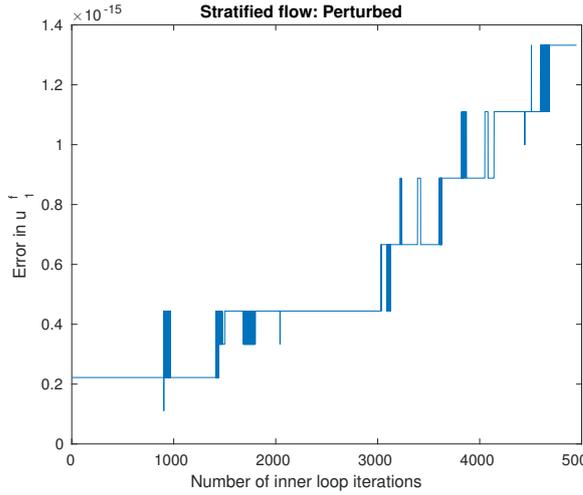


Figure 10.37: Stratified flow Perturbed case: Plot of L^∞ norm of the error in u_f^1

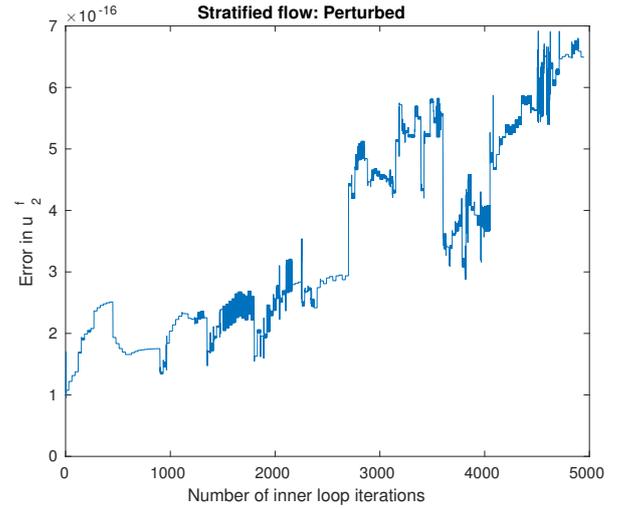


Figure 10.38: Stratified flow Perturbed case: Plot of L^∞ norm of the error in u_f^2

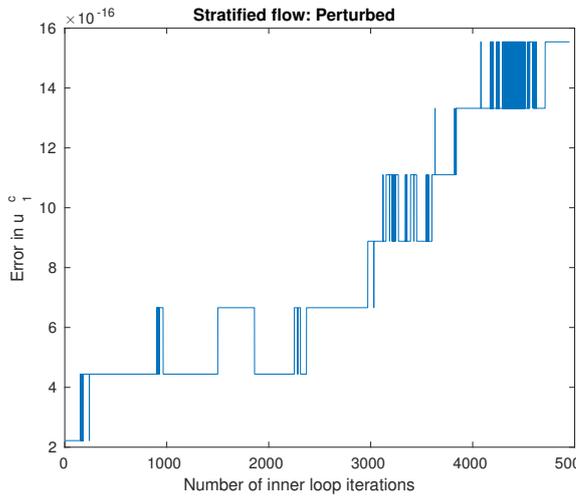


Figure 10.39: Stratified flow Perturbed case: Plot of L^∞ norm of the error in u_1^c

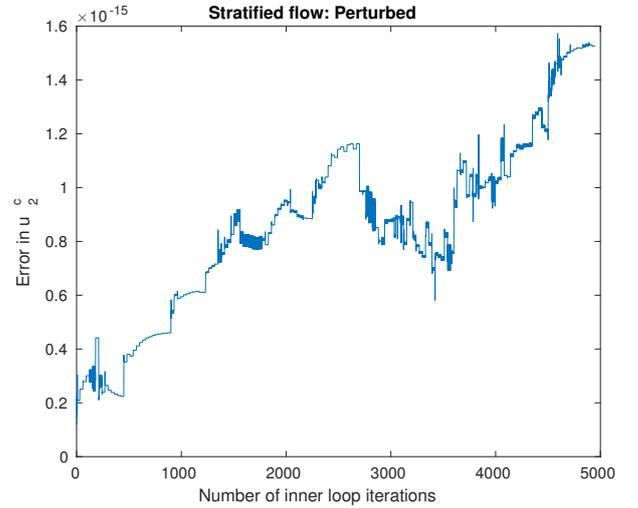


Figure 10.40: Stratified flow Perturbed case: Plot of L^∞ norm of the error in u_2^c

10.3.2. PERIODIC BOUNDARY CONDITIONS

Periodic boundary conditions mimic an infinite channel, which is useful for assessing fully developed flows such as the steady state channel flow. The following boundary conditions are imposed for the case of periodic boundaries:

- **Top and Bottom Boundary**
 - Pressure- Non-homogeneous Neumann
 - Velocity in \hat{e}_1 - Homogeneous Neumann
 - Velocity in \hat{e}_2 - Homogeneous Dirichlet
- **Left and Right Boundary**
 - Pressure - Periodic

- Velocity in \hat{e}_1 - Periodic
- Velocity in \hat{e}_2 - Periodic

Figures 10.41 to 10.44 show the L^∞ norm of the error in the velocities as a function of the inner loop iterations.

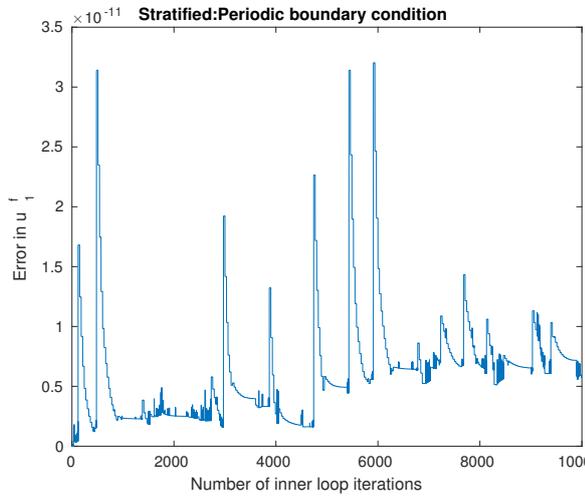


Figure 10.41: Stratified flow Periodic boundaries:
Plot of L^∞ norm of the error in u_f^1

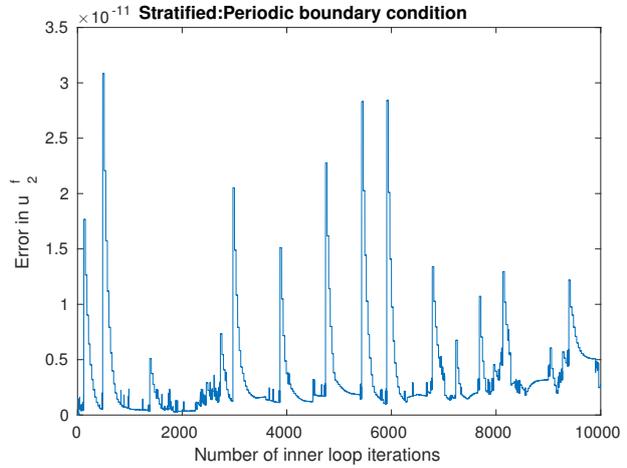


Figure 10.42: Stratified flow Periodic boundaries:
Plot of L^∞ norm of the error in u_f^2

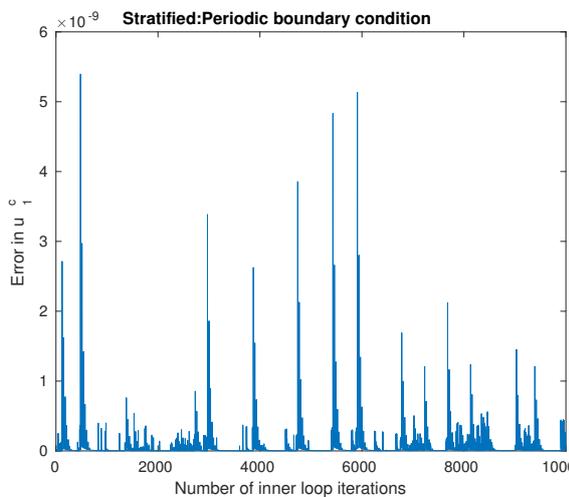


Figure 10.43: Stratified flow Periodic boundaries:
Plot of L^∞ norm of the error in u_1^c

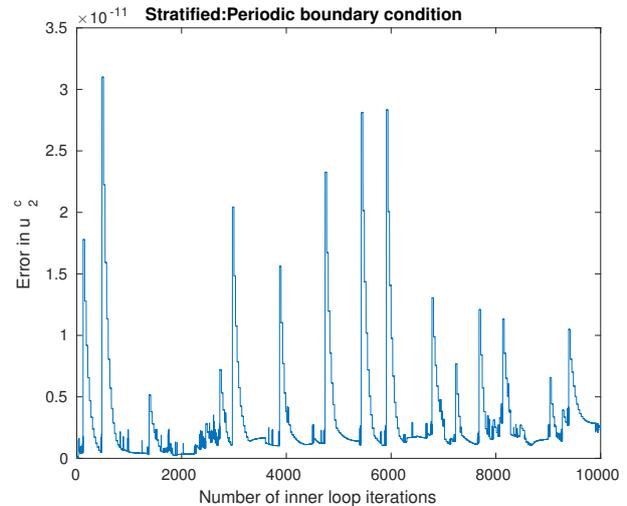


Figure 10.44: Stratified flow Periodic boundaries:
Plot of L^∞ norm of the error in u_2^c

10.3.3. ANALYSIS

The visualizations of the error in the face and cell center velocity show an increase in the errors over the course of the iterations, in contrast to the closed box case, where the errors stabilize and level off. Although a force balance is produced, inferred from the fact that the numerical solution obtained in the first inner loop iteration is accurate upto machine precision, the error in the solution slowly builds up over time. Despite the fact that the flow variables are initialized to the exact analytical solution, small round off errors build up and destabilize the discrete system, pointing to an unstable equilibrium. This can most likely be attributed to two factors:

- floating point arithmetic
- unstable nature of the operators

In the following section, these two factors are investigated and analyzed.

FLOATING POINT ARITHMETIC

The correction calculated for the flow variables is small and comparable to machine precision at the beginning of the simulation. Hence, the corrected values of the flow variables can be affected by rounding off errors of comparable magnitude as the correction. Although the pressure correction is of comparable size to the round of error, it is essential to make the face velocity field divergence free. Furthermore, the magnitude of the pressure is higher than the magnitude of the correction in pressure. Hence, due to a limited computational precision, when the pressure is corrected, the mantissa of corrected pressure cannot account for the mantissa of the correction in pressure.

To counter this limitation, the pressure is split into a background pressure and the perturbation over the background pressure. The operators are evaluated individually for the split values. Furthermore, the perturbation is updated with the correction in pressure to ensure the correction is not lost due to limited numerical precision.

The pressure is split as the background pressure \bar{p} and perturbation over this pressure \dot{p}

$$p = \bar{p} + \dot{p}. \quad (10.11)$$

The operators are evaluated individually

$$Bp = B\bar{p} + B\dot{p}, \quad (10.12)$$

$$DG^f p = DG^f \bar{p} + DG^f \dot{p}. \quad (10.13)$$

Furthermore, the pressure is updated by correcting the perturbation. Hence,

$$\dot{p}^{l+1} = \dot{p}^l + \delta p. \quad (10.14)$$

SINGLE PHASE FLOW: BEHAVIOR OF DISCRETE OPERATORS

It is sufficient for a stationary discrete formulation to comply with the differential equation and the interface jump conditions. However, to apply the approach to solve unsteady flow problems the influence of the reformulated operators on the stability properties of the system has to be investigated. In this subsection the properties of the individual operators and their behavior when they are embedded in the complete solver algorithm is analyzed for single phase flow. Based on the analysis of the operators for single phase flow, the nature of the behavior of the operators is extrapolated to the case for multiphase flow with a large density ratio.

Conservation of momentum in discrete form is written as

$$\frac{du}{dt} + C(u)u = -Gp + Ju \quad (10.15)$$

where $C(u)$ is the divergence of the linearized convective flux function. The discrete form of the continuity equation is written as

$$Du + Bp = 0 \quad (10.16)$$

The temporal rate of change in the kinetic energy in the domain is expressed as

$$\frac{1}{2} \frac{du^T u}{dt} = \frac{1}{2} u^T \frac{du}{dt} + \frac{1}{2} \frac{du^T}{dt} u \quad (10.17)$$

Taking the transpose of (10.14) gives

$$\frac{du^T}{dt} + u^T C^T(u) = -p^T G^T + u^T J^T \quad (10.18)$$

Pre-multiplying equation (10.14) by u^T and post multiplying (10.18) by u and substituting in 10.17 gives

$$\frac{1}{2} \frac{du^T u}{dt} = -\frac{1}{2} (u^T (C(u) + C^T(u)) u) + \frac{1}{2} (u^T (J + J^T) u) - \frac{1}{2} (u^T Gp + p^T G^T u) \quad (10.19)$$

As the convection terms are discretized using a central scheme the operator C is skew symmetric. Hence

$$\frac{1}{2} \frac{du^T u}{dt} = \frac{1}{2} (u^T (J + J^T) u) - \frac{1}{2} (u^T G p + p^T G^T u) \quad (10.20)$$

It holds that

$$D^T = -G \quad (10.21)$$

Hence, using (10.21) and (10.16) in (10.20), gives

$$\begin{aligned} \frac{1}{2} \frac{du^T u}{dt} &= \frac{1}{2} (u^T (J + J^T) u) - \frac{1}{2} (p^T B^T p + p^T B p) \\ &= \frac{1}{2} (u^T (J + J^T) u) - \frac{1}{2} (p^T (B^T + B) p) \\ &= u^T J u + p^T B p \end{aligned} \quad (10.22)$$

In the above equation, it can be seen that if the operator J is symmetric negative definite then

$$\forall u \in \mathbb{R} \setminus \{0\} : u^T J u < 0. \quad (10.23)$$

Similarly, if B is symmetric positive definite then

$$\forall p \in \mathbb{R} \setminus \{0\} : p^T B p > 0. \quad (10.24)$$

Hence,

$$\frac{1}{2} \frac{du^T u}{dt} = \frac{1}{2} (u^T (J + J^T) u) - \frac{1}{2} (p^T (B^T + B) p) < 0 \quad (10.25)$$

It can be inferred that the kinetic energy of a perturbation will decrease if J is symmetric negative definite and B is symmetric positive definite. Hence, if the numerical stability conditions are met, the resulting system will be stable and can be numerically integrated in a stable manner with a time-integration method.

ANALYSIS OF MULTIPHASE FLOWS

Pressure splitting was employed to evaluate the operators and the pressure update in the integrated solver, as explained in Section 10.3.3. The numerical solution was indistinguishable for the implementation with and without the splitting. Hence, it can be concluded that the characteristic of an unstable equilibrium is not likely explainable by a limited numerical precision.

The stratified two-phase flow has a uniform velocity and an exact analytical solution dictated by the continuous form of the governing equations. The discrete formulation satisfies the interface and the boundary conditions, hence it can be inferred that a solution to the discrete system exists, namely the exact solution with which the solver is initialized. Analysis of the evolution of discrete kinetic energy for single phase flows pointed to the stability relations the operators should hold. It can be reasonably speculated that the stability constraints should hold for operators constructed for multiphase flows. The eigenspectrum of the formulated operator B was studied, wherein an inhomogeneous Dirichlet boundary condition is imposed for pressure at the top boundary. It was noted that the eigenspectrum for this case contains both negative and positive eigenvalues. The spectrum of a symmetric (semi) positive definite matrix is strictly (non-negative) positive. Hence, it can be inferred that the formulated B operator, for the case of stratified flow is not a symmetric positive definite matrix.

Stratified two-phase flow is an exceptional case, where the diffusion term vanishes due to uniform velocity in the computational domain. As a result the contribution of the operator J is small compared to the contribution of the operator B . Hence over the course of the iterations, a (unavoidable) non-zero round off error develops in the vector projection of the pressure gradient in the direction of the negative valued eigenmode. As a result the kinetic energy increases as the contribution of the operator B overwhelms the contribution of the operator J .

Furthermore, the formulation of the operator B leads to a shift in the weights of the contributions of the surrounding control volumes creating a non-standard stencil (listed in Appendix-A). The effect of such a shift in the weights is unclear and needs to be investigated.

11

CONCLUSIONS

The objective of this research was to formulate a balanced-force numerical framework applicable for incompressible, immiscible two-phase flows for the case where the interface between the two fluids does not have to coincide or to be aligned with the faces of the control volumes in the computational domain. This was achieved by formulating a reconstruction of the flow variables using a second order Taylor expansion at the interface. Discrete operators were formulated, in which the gradients of the flow variables are evaluated in the limit values of the gradient upon approaching the interface. Numerical experiments were carried out to ascertain the efficacy of the formulation for a quiescent flow for the case where the interface coincides, is offset and aligned and is arbitrarily inclined to the faces of the control volumes. The accuracy of the formulation was studied by comparing the order of the local truncation error computed mathematically and numerically. An iterative in-stationary solver for multiphase flows based on the approach of Miller and Schmidt was proposed. The numerical framework was analyzed for steady and unsteady dynamic cases of quiescent flow, a viscous standing wave in a closed tank, dam break and stratified flow. Finally, an examination of the properties and the behavior of the operators embedded in the iterative solver was carried out.

CONCLUSIONS

Based on the extensive analysis of the formulated numerical method and the numerical experiments performed the following conclusions can be drawn about the balanced-force formulation to numerically model two-phase incompressible, immiscible flows:

- A balanced-force framework applicable for interface that is arbitrarily located and oriented with respect the faces of the control volumes is achieved by:
 - Reconstruction of the flow variables at the interface based on the interface jump conditions.
 - Discretization of the gradients in the discrete operators in the limit of approaching the interface such that the operators comply with the interface jump conditions.
 - Employing a balanced-force based density.

The numerical solution produced by this framework avoids the occurrence of spurious velocities and leads to a physically consistent solution.

- The local truncation error in the operators B and DG^f is fourth order accurate away from the interface and second order accurate near the interface. Hence the constructed operators do not lead to a loss of accuracy of the discrete system.
- The interface jump condition for the pressure simplifies to the hydrostatic interface jump condition for the case where the Reynolds number tends to infinity. The numerical experiments of the dynamic cases carried out at low Reynolds number also gave accurate results, which implies the hydrostatic interface jump condition is a sufficient condition to model two-phase flows at low and high Reynolds numbers.

- For the case of unsteady flows, wherein the discrete operators are incorporated in the iterative solver stability relations need to hold true. In exceptional cases where the diffusion term vanishes, the destabilizing contribution of the operator B overwhelms the stabilizing contribution of the operator J , likely leading to a discrete system with an unstable equilibrium.

RECOMMENDATION FOR FUTURE RESEARCH

In this research, the coefficients of the pressure gradients in the operators involved in the the formulation of the face velocity and the pressure-velocity coupling were scaled by the inverse of the density and the time step to ensure dimensional correctness. The individual derivatives were calculated such that the vector projection of the pressure gradient in the direction of gravity complies with the hydrostatic interface jump condition. The coefficients can also be formulated to take into account the convective and the diffusive term as elaborated by Miller and Schmidt [13] and Rhie and Chow [11]. The precise choice of the coefficients cannot be mathematically justified, and hence a sensitivity analysis to the choice of coefficients is required.

The nature of the local truncation errors in the operators, which are integral to the balanced-force formulation, was investigated. Further analysis could be done regarding the characteristics of the global truncation error of the discrete system.

The proposed formulations were implemented on a Cartesian grid employing the collocated arrangement of unknowns. The collocated arrangement of unknowns provides an opportune choice to discretize the system of equations on an unstructured computational domain. Hence, an effort could be taken to extend the formulation to general arbitrary unstructured control volumes.

In this research the level set method was utilized as an interface tracking method. A drawback of this method is that it requires reinitialization to maintain the property of a signed distance function which leads to a loss of mass over the course of time steps. The balanced-force formulation requires the location and the orientation of the interface to construct the operators involved in the framework. Hence an approach to employ this formulation coupled with another interface tracking method that can provide the necessary information, such as the Geometric Volume of Fluid method, could be investigated.

Stability relations for the operators were derived for single phase flows. Based on these relations, the stability requirements for the operators in multiphase flows were conjectured. As the stability requirements for the balanced-force formulation only holds true when the system reduces to single phase flows, a more detailed concrete investigation is needed to draw definite conclusions.

The conducted numerical experiments conducted displayed the conformance of the formulation with the interface jump conditions. The balanced-force numerical framework can be used to validate flow instabilities that require such a sharp interface treatment. For example the lateral roll of ship creates waves at the water-air interface. A balanced-force numerical model will ensure that the numerical solution produces physically consistent waves without adding any numerical dispersion. The proposed framework can be integrated in an available flow solver, such as OpenFOAM, to evaluate more complex dynamic flow cases.

"The more I read, the more I acquire, the more certain I am that I know nothing".

— Voltaire

A

APPENDIX-A

The stencils obtained for a computational domain of size $l \times l$ for the discrete operators DG^f and B are listed below for the following computational and fluid properties:

Computational Properties

- Mesh width $\frac{h}{7}$: 0.01
- Interface location : $\frac{x_2}{7} = 0.5$

Computational Properties

- Density : $\rho_1 = 50, \rho_2 = 1$

The stencils have been listed for the control volumes located near the interface as shown in Figure A.1 and A.2, for operator B and DG^f , respectively. A comparison of the resulting stencil is made between single phase flow of density $\rho = 50$, single phase flow of density $\rho = 1$ and two-phase flow of $\rho_1 = 50$ and $\rho_2 = 1$.

A.1. STENCILS OF OPERATOR B

CV	$\rho = 50$	$\rho = 1$	$\rho_1 = 50, \rho_2 = 1$
1	$\begin{bmatrix} 0.05 \\ -0.2 \\ 0.05 \ -0.2 \ 0.6 \ -0.2 \ 0.05 \\ -0.2 \\ 0.05 \end{bmatrix}$	$\begin{bmatrix} 2.5 \\ -10 \\ 2.5 \ -10 \ 30 \ -10 \ 2.5 \\ -10 \\ 2.5 \end{bmatrix}$	$\begin{bmatrix} 2.5 \\ -2.7941 \\ 0.05 \ -0.02 \ 0.7441 \ -0.02 \ 0.05 \\ -0.02 \\ 0.05 \end{bmatrix}$
2	$\begin{bmatrix} 0.05 \\ -0.2 \\ 0.05 \ -0.2 \ 0.6 \ -0.2 \ 0.05 \\ -0.2 \\ 0.05 \end{bmatrix}$	$\begin{bmatrix} 2.5 \\ -10 \\ 2.5 \ -10 \ 30 \ -10 \ 2.5 \\ -10 \\ 2.5 \end{bmatrix}$	$\begin{bmatrix} 0.0980 \\ -0.2480 \\ 0.05 \ -0.02 \ 0.6 \ -0.02 \ 0.05 \\ -0.02 \\ 0.05 \end{bmatrix}$
3	$\begin{bmatrix} 0.05 \\ -0.2 \\ 0.05 \ -0.2 \ 0.6 \ -0.2 \ 0.05 \\ -0.2 \\ 0.05 \end{bmatrix}$	$\begin{bmatrix} 2.5 \\ -10 \\ 2.5 \ -10 \ 30 \ -10 \ 2.5 \\ -10 \\ 2.5 \end{bmatrix}$	$\begin{bmatrix} 2.5 \\ -10 \\ 2.5 \ -10 \ 30 \ -10 \ 2.5 \\ -7.5980 \\ 0.0980 \end{bmatrix}$
4	$\begin{bmatrix} 0.05 \\ -0.2 \\ 0.05 \ -0.2 \ 0.6 \ -0.2 \ 0.05 \\ -0.2 \\ 0.05 \end{bmatrix}$	$\begin{bmatrix} 2.5 \\ -10 \\ 2.5 \ -10 \ 30 \ -10 \ 2.5 \\ -10 \\ 2.5 \end{bmatrix}$	$\begin{bmatrix} 2.5 \\ -10 \\ 2.5 \ -10 \ 22.7941 \ -10 \ 2.5 \\ -0.3441 \\ 0.05 \end{bmatrix}$

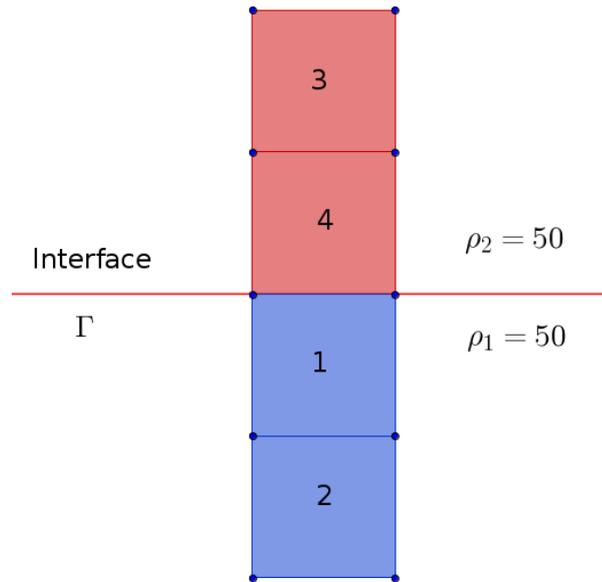


Figure A.1: Location of the control volumes: Operator B

A.2. STENCIL OF OPERATOR DG^f

CV	$\rho = 50$	$\rho = 1$	$\rho_1 = 50, \rho_2 = 1$
1	$\begin{bmatrix} -1 & & \\ -1 & 4 & -1 \\ & -1 & \end{bmatrix}$	$\begin{bmatrix} & -0.2 & \\ -0.2 & 0.8 & -0.2 \\ & -0.2 & \end{bmatrix}$	$\begin{bmatrix} & -0.392 & \\ -0.2 & 0.992 & -2 \\ & -0.2 & \end{bmatrix}$
2	$\begin{bmatrix} & -1 & \\ -1 & 4 & -1 \\ & -1 & \end{bmatrix}$	$\begin{bmatrix} & -0.2 & \\ -0.2 & 0.8 & -0.2 \\ & -0.2 & \end{bmatrix}$	$\begin{bmatrix} & -1 & \\ -1 & 3.0392 & -1 \\ & -0.0392 & \end{bmatrix}$

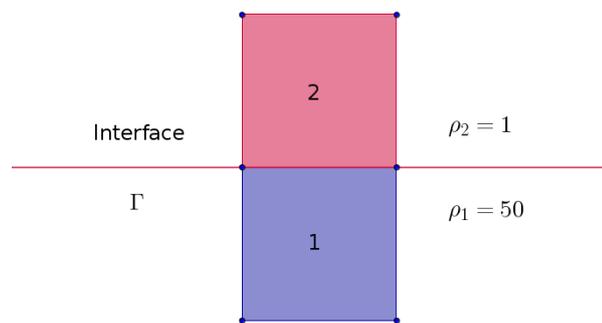


Figure A.2: Location of the control volumes: Operator DG^f

B

APPENDIX-B

A comparison was made between the numerical solution obtained for the proposed balanced-force formulation and the numerical solution of OpenFOAM(Version 2.4.0). The files specifying the OpenFOAM setup have been listed below. FunkySetFields utility, part of swak4Foam was utilized to initialize the perturbed interface between the two fluids.

```
/*-----* C++ -*-----*\
| ===== |
| \\      / F i e l d      | OpenFOAM: The Open Source CFD Toolbox |
| \\      / O p e r a t i o n | Version: 2.4.0 |
| \\      / A n d           | Web:      www.OpenFOAM.org |
| \\      / M a n i p u l a t i o n |
\*-----*/
FoamFile
{
    version      2.0;
    format       ascii;
    class        volScalarField;
    object       p;
}
// *****
dimensions      [1 -1 -2 0 0 0];
internalField   uniform 0;
boundaryField
{
    wall
    {
        type      empty;
    }

    inlet1
    {
        type      zeroGradient;
    }

    outlet1
    {
        type      zeroGradient;
    }

    wall_upper
    {
        type      zeroGradient;
    }

    wall_lower
    {
        type      zeroGradient;
    }
}
// *****
```

```

/*-----* C++ *-----*/
|=====|
| \ \ / / F i e l d | OpenFOAM: The Open Source CFD Toolbox |
| \ \ / / O p e r a t i o n | Version: 2.4.0 |
| \ \ / / A n d | Web: www.OpenFOAM.org |
| \ \ / / M a n i p u l a t i o n |
|-----*/
FoamFile
{
    version      2.0;
    format       ascii;
    class        volVectorField;
    object       U;
}
// ***** //

dimensions      [0 1 -1 0 0 0 0];

internalField   uniform (0 0 0);

boundaryField
{
    wall
    {
        type      empty;
    }

    wall_upper
    {
        type      slip;
    }

    wall_lower
    {
        type      slip;
    }

    inlet1
    {
        type      slip;
    }

    outlet1
    {
        type      slip;
    }
}
// ***** //

```

```
/*-----* C++ *-----*/
|=====|
| \\ / F i e l d | OpenFOAM: The Open Source CFD Toolbox |
| \\ / O p e r a t i o n | Version: 2.4.0 |
| \\ / A n d | Web: www.OpenFOAM.org |
| \\ / M a n i p u l a t i o n | |
/*-----*/

FoamFile
{
    version      2.0;
    format       ascii;
    class        volScalarField;
    location     "0";
    object       alpha.water;
}
// ***** //

dimensions      [0 0 0 0 0 0 0];
internalField   uniform 0;

boundaryField
{
    boundaryField
    {
        outlet1
        {
            type            zeroGradient;
        }
        inlet1
        {
            type            zeroGradient;
        }
        wall_upper
        {
            type            zeroGradient;
        }
        wall_lower
        {
            type            zeroGradient;
        }
        wall
        {
            type            empty;
        }
    }
}
}
```

```

/*-----* C++ *-----*/
| ===== | |
| \\ / F i e l d | OpenFOAM: The Open Source CFD Toolbox |
| \\ / O p e r a t i o n | Version: 1.6 |
| \\ / A n d | Web: www.OpenFOAM.org |
| \\ / M a n i p u l a t i o n | |
\-----*/
FoamFile
{
    version      2.0;
    format       ascii;
    class        dictionary    location    "system";
    object       controlDict;
}
// ***** //

application    interFoam;

startFrom      latestTime;

startTime      0.0;

stopAt         endTime;

endTime        20;

deltaT         0.0005;

writeControl   adjustableRunTime;

writeInterval  0.02;

purgeWrite     0;

writeFormat    ascii;

writePrecision 6;

writeCompression  uncompressed;

timeFormat     general;

timePrecision  6;

runTimeModifiable yes;

adjustTimeStep yes;

maxCo          0.5;
maxAlphaCo     0.05;

maxDeltaT      1;

// ***** //

```

```

/*-----* C++ *-----*/
|=====|
| \\ / F i e l d | OpenFOAM: The Open Source CFD Toolbox |
| \\ / O p e r a t i o n | Version: 2.4.0 |
| \\ / A n d | Web: uuv.OpenFOAM.org |
| \\ / M a n i p u l a t i o n | |
/*-----*/
FoamFile
{
    version      2.0;
    format       ascii;
    class        dictionary;
    location     "system";
    object       fvSchemes;
}
// ***** //

ddtSchemes
{
    default      Euler;
}

gradSchemes
{
    default      Gauss linear;
}

divSchemes
{
    div(rhoPhi,U) Gauss linearUpwind grad(U);
    div(phi,alpha) Gauss vanLeer;
    div(phirb,alpha) Gauss linear;
    div((muEff*dev(T(grad(U)))) Gauss linear;
}

laplacianSchemes
{
    default      Gauss linear corrected;
}

interpolationSchemes
{
    default      linear;
}

snGradSchemes
{
    default      corrected;
}

fluxRequired
{
    default      no;
    p_rgh;
    pcorr;
}
// ***** //

```

```

/*-----* C++ *-----*/
|=====|
| \ \ / F i e l d | OpenFOAM: The Open Source CFD Toolbox |
| \ \ / O p e r a t i o n | Version: 2.4.0 |
| \ \ / A n d | Web: www.OpenFOAM.org |
| \ \ / M a n i p u l a t i o n |
|-----*/
FoamFile
{
    version      2.0;
    format       ascii;
    class        dictionary;
    location     "system";
    object       fvSolution;
}
// ***** //

solvers
{
    "alpha.water.*"
    {
        nAlphaCorr      1;
        nAlphaSubCycles 10;
        cAlpha          1;
    }

    p_rgh
    {
        solver          GAMG;
        tolerance       1e-08;
        relTol          0.01;
        smoother        DIC;
        nPreSweeps      0;
        nPostSweeps     2;
        nFinestSweeps   2;
        cacheAgglomeration false;
        nCellsInCoarsestLevel 10;
        agglomerator     faceAreaPair;
        mergeLevels     1;
    }

    p_rghFinal
    {
        $p_rgh;
        relTol          0;
    }

    "pcorr.*"
    {
        $p_rghFinal;
        tolerance       0.0001;
    }

    U
    {
        solver          smoothSolver;
        smoother        GaussSeidel;
        tolerance       1e-06;
        relTol          0;
        nSweeps         1;
    }

    "(k|B|nuTilda)"
    {
        solver          smoothSolver;
        smoother        symGaussSeidel;
        tolerance       1e-08;
        relTol          0;
    }
}

```

PIMPLE

```
{
    momentumPredictor no;
    nCorrectors      3;
    nNonOrthogonalCorrectors 0;

    pRefPoint      (0.51 0.51 0);
    pRefValue      0;
}

// ***** //
```


BIBLIOGRAPHY

- [1] P. Queutey and M. Visonneau, *An interface capturing method for free-surface hydrodynamic flows*, *Computers and Fluids* **36**, 1481 (2007).
- [2] C. E. Brennen, *Fundamentals of Multiphase Flows* (2005).
- [3] M. Kang, R. P. Fedkiw, and X. D. Liu, *A boundary condition capturing method for multiphase incompressible flow*, *Journal of Scientific Computing* **15**, 323 (2000).
- [4] S. Osher and R. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces* (2006).
- [5] A. Prosperetti and G. Tryggvason, *Computational Methods for multiphase flows* (Cambridge, 2007).
- [6] C. W. Hirt and B. D. Nichols, *Volume of fluid (VOF) method for the dynamics of free boundaries*, *Journal of Computational Physics* **39**, 201 (1981), arXiv:0924-0136(97)00224-0 .
- [7] R. Scardovelli and G. Tryggvason, *Direct Numerical Simulations of Gas – Liquid Multiphase Flows*.
- [8] V. R. Gopala and B. G. M. van Wachem, *Volume of fluid methods for immiscible-fluid and free-surface flows*, *Chemical Engineering Journal* **141**, 204 (2008).
- [9] F. Denner and B. G. M. van Wachem, *Compressive VOF method with skewness correction to capture sharp interfaces on arbitrary meshes*, *Journal of Computational Physics* **279**, 127 (2014).
- [10] O. Ubbink and R. Issa, *A Method for Capturing Sharp Fluid Interfaces on Arbitrary Meshes*, *Journal of Computational Physics* **153**, 26 (1999).
- [11] W. Rhie, C.M and Chow, “*Numerical Study of the Turbulent Flow Past an Airfoil with Trailing Edge Separation.*”, *AIAA Journal* 21 **11**, 1525 (1983).
- [12] F. Denner and B. G. M. van Wachem, *Fully-Coupled Balanced-Force VOF Framework for Arbitrary Meshes with Least-Squares Curvature Evaluation from Volume Fractions*, *Journal of Computational Physics* **213**, 218 (2014), arXiv:1405.0829 .
- [13] T. F. Miller and F. W. Schmidt, *Use of a Pressure-Weighted Interpolation Method for the Solution of the Incompressible Navier-Stokes Equations on a Nonstaggered Grid System*, *Numerical Heat Transfer* **14**, 213 (1988).
- [14] M. M. Francois, S. J. Cummins, E. D. Dendy, D. B. Kothe, J. M. Sicilian, and M. W. Williams, *A balanced-force algorithm for continuous and sharp interfacial surface tension models within a volume tracking framework*, *Journal of Computational Physics* (2006), 10.1016/j.jcp.2005.08.004.
- [15] J. Mencinger and I. Žun, *On the finite volume discretization of discontinuous body force field on collocated grid: Application to VOF method*, *Journal of Computational Physics* **221**, 524 (2007).
- [16] S.-K. Choi, S.-O. Kim, C.-H. Lee, and H.-K. Choi, *Use of the Momentum Interpolation Method for Flows With a Large Body Force*, *Numerical Heat Transfer, Part B: Fundamentals* **43**, 267 (2003).
- [17] R. Wemmenhove, *Numerical simulation of two-phase flow in offshore environments* (2008).
- [18] H. Rusche, *Computational Fluid Dynamics of Dispersed Two-Phase Flows at High Phase Fractions*, PhD Thesis **1**, 335 (2002).
- [19] S. S. Deshpande, L. Anumolu, and M. F. Trujillo, *Evaluating the performance of the two-phase flow solver interFoam*, *Computational Science & Discovery* **5**, 014016 (2012).
- [20] R. Fedkiw and X. Liu, *The ghost fluid method for viscous flows*, *Progress in Numerical Solutions of Partial Differential ...* , 1 (1998).

-
- [21] R. P. Fedkiw, T. Aslam, B. Merriman, and S. Osher, *A Non-oscillatory Eulerian Approach to Interfaces in Multimaterial Flows (the Ghost Fluid Method)*, *Journal of Computational Physics* **152**, 457 (1999), arXiv:jcph.1999.6236 .
- [22] P. Wesseling, *Principles of Computational Fluid Dynamics* (Springer).
- [23] G. M. Peric, R. Kessler, *Comparison of finite-volume numerical methods with staggered and collocated grids*, *Computers & Fluids* **16**, 389 (1988).
- [24] D. M. Greaves, *Viscous waves and wave-structure interaction in a tank using adapting quadtree grids*, *Journal of Fluids and Structures* **23**, 1149 (2007).
- [25] G. X. Wu, R. Eatock Taylor, and D. M. Greaves, *The effect of viscosity on the transient free-surface waves in a two-dimensional tank*, *Journal of Engineering Mathematics* **40**, 77 (2001).