AMERICAN UNIVERSITY OF BEIRUT

The Development of a Fully Coupled Solver for the Solution of Free Surface Flows

by
MHAMAD MAHDI ALLOUSH

A thesis
submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Engineering
to the Department of Mechanical Engineering
of Maroun Semaan Faculty of Engineering and Architecture
at the American University of Beirut

Beirut, Lebanon
October 2020

AMERICAN UNIVERSITY OF BEIRUT

The Development of a Fully Coupled Solver for the Solution of Free Surface Flows

by

MHAMAD MAHDI ALLOUSH

Approved by:

Dr. Marwan Darwish, Professor — Advisor

Mechanical Engineering

Marwan

Dr. Luca Mangani, Professor — Co-Advisor

Mechanical Engineering

Dr. Fadl Moukalled, Professor — Member of Committee

Mechanical Engineering

Moukalled Fadl

Dr. Nesreene Ghaddar, Professor — Member of Committee

Mechanical Engineering

N. Ghad

Dr. Kamel Aboughali, Professor — Member of Committee

Mechanical Engineering

Dr. Nabil Nassif, Professor Member of Committee

Mathematics

Nabil R. Nassif

Dr. Abdellah Kharicha, Professor Member of Committee

Metallurgy

Date of thesis defense: October 30, 2020

AMERICAN UNIVERSITY OF BEIRUT

THESIS, DISSERTATION, PROJECT RELEASE FORM

Student Name: Alloush Mhamad Mahdi Hani
Last First Middle

○ Master's Thesis ○ Master's Project ⊗ Doctoral Dissertation

☒ I authorize the American University of Beirut to: (a) reproduce hard or electronic copies of my thesis, dissertation, or project; (b) include such copies in the archives and digital repositories of the University; and (c) make freely available such copies to third parties for research or educational purposes.

☐ I authorize the American University of Beirut, to: (a) reproduce hard or electronic copies of it; (b) include such copies in the archives and digital repositories of the University; and (c) make freely available such copies to third parties for research or educational purposes after: **One _____ year from the date of submission of my thesis, dissertation or project.**

Two _____ years from the date of submission of my thesis , dissertation or project.
Three _____ years from the date of submission of my thesis , dissertation or project.

Mahdi 03/11/2020
Signature Date

This form is signed when submitting the thesis, dissertation, or project to the University Libraries

Acknowledgments

I would like to thank my advisor Professor Marwan Darwish who has largely contributed to this research work, and my co-advisor Professor Luca Mangani from Lucerne University of Applied Sciences who has also provided supervision and guidance throughout the journey. Special thanks go to Professor Fadl Moukalled for his time and effort which he put to make this task accomplished. I would also like to thank the other academic board members and colleagues in the CFD labs at the American University of Beirut and Lucerne University of Applied Sciences who cooperated and contributed to this work. It was also a pleasure for me to have Ms. Jana Ftouni a part of the team, devoting her time to review and edit this thesis report linguistically. Last and not least, I highly appreciate the effort, the patience, and the support of my family members who have motivated me towards finalizing my PhD trip over the past five years.

An Abstract of the Thesis of

Mhamad Mahdi Alloush for PhD of Engineering
Major: Mechanical Engineering

Title: The Development of a Fully Coupled Solver for the Solution of Free Surface Flows

In this thesis, the development of a fully coupled finite volume based free surface flow solver is presented in details. The approach is based on the volume of fluid method (VOF) but extends it to account for a range of couplings, including velocity-pressure, velocity-volume fraction, volume fraction-velocity and volume fraction-pressure couplings. This results in a matrix of coefficients formed at each control volume. Accounting for the various couplings leads to an improved convergence and robustness as compared to segregated type solvers, especially for steady-state applications but also for transient free surface flows. The solver is additionally equipped with a number of techniques, including the treatment of the body forces at the free surface to guarantee a force-balanced solution; this treatment is evaluated analytically and numerically. Also, a very important facet to ensure the geometric constraint is provided in the solver; this is represented by the summation of all volume fractions being enforced to one. In this thesis, a novel implicit approach to enforce this constraint is presented and is compared to standard approaches currently in use. Compressive schemes are also addressed in this work and incorporated in the solver in order to capture the interface with high accuracy and fidelity. The solver also handles turbulence modeling infrastructure for turbulent free surface flows with a set of two- and four-equation models. Worth mentioning is that the performance of the coupled VOF solver is compared to that of a very popular commercial solver which also features a coupled VOF solver. On the other hand, accuracy of the results is evaluated in the light of experimental data along with data generated by the commercial solver. A range of free surface problems exhibiting a variety of boundary conditions and different levels of complexity are attempted. A transient 2D forced sloshing test case is first simulated and the progress of the free surface at a predefined point is monitored in time; it shows consistent results with experimental data. A 3D dam break problem, a very popular free surface testing benchmark, is also simulated;

it also produces results comparable to experimental data. Two other steady-state test cases featuring open channel flows past a cylinder and a submerged hydrofoil respectively are simulated. Drag monitors for the two cases show a competitive convergence rate as compared to the commercial software.

Contents

List of Figures

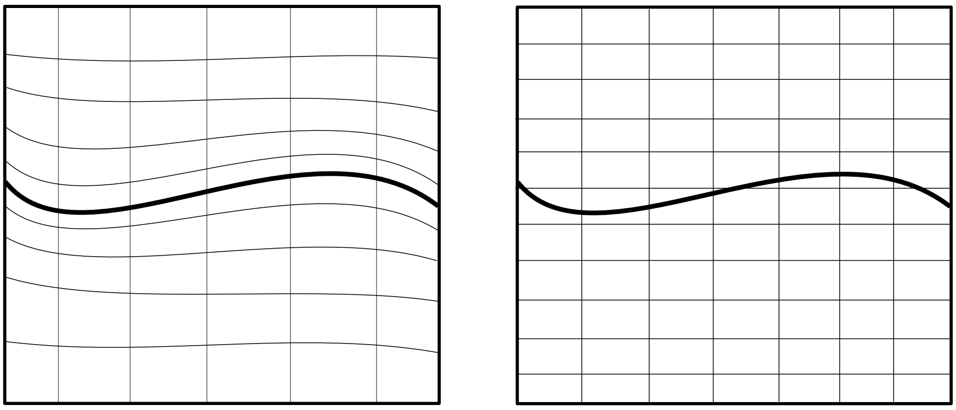
Chapter 1

Introduction

1.1 Free Surface Flows in Engineering

Developing more sophisticated tools to simulate complex engineering problems sits at the top of many research and development efforts in the area of computational fluid dynamics (CFD). In particular, the use of numerical simulations in the area of ship design, metal casting or any other domain which involves separated phases, has pushed many academic authorities to develop more efficient and robust numerical methods in the field of free surface flows [1, 2, 3, 4]. In free surface flows, it is required to solve for each of the phases as well as the interface among them, and hence, the simulation of free surface flows necessitates an appropriate treatment of the interface capturing method used in order to credibly preserve the sharpness of the interface. There have been strong efforts to establish numerical techniques to capture the interface because the free surface is free to deform due to the nature of the flow, the fact which requires more than one model to capture the interface. Numerical solutions to the aforementioned free surface problem can be made using two meshes, a Lagrangian mesh or an Eulerian mesh. In the Lagrangian method, the grid moves with the interface such that its boundary is always aligned to it (see figure 1.1a). The interface in this method is precisely followed, but however, there's a drawback related to the distortion of the mesh, which produces instability to the solver. Eulerian meshes (figure 1.1b) are stationary where the fluids travel in the fixed mesh. This method does not suffer from the element distortion problem as in the Lagrangian mesh. However, using the Eulerian mesh necessitates the solving of an additional scalar equation to track the interface. The details of the different methods are addressed by [5].

Among the techniques which are based on the Eulerian approach, is the fixed grid technique where the fluid interface is captured directly. The first fixed grid technique is the marker-and-cell approach by Harlow and Welch [6]. It introduces massless particles in the fluid domain, of velocities determined from interpolated velocities, and the cells that contain particles are considered to have one phase in


(a) Lagrangian mesh

(b) Eulerian mesh

Figure 1.1: Two types of free surface meshes which refer to two different approaches, the Lagrangian approach and the Eulerian approach

them, otherwise, they don't include this phase. The method was later extended by [7] to track particles on the surface only. Hirt and Nichols [8] later developed the volume-of-fluid (VOF) method, and it is one of the most popular interface capturing techniques, where the phases are assumed to share the velocity, pressure and temperature fields, while the transport properties (density, viscosity, etc.) are being changed in space and time. This method can be adapted for any type of mesh, and for an arbitrary number of phases [9, 10]. The VOF method also assumes that the phases are immiscible with a clearly defined interface. Typical problems solved with the VOF method are motion of liquid in confined spaces, open channel flows, and steady or transient tracking of any liquid-gas interface. It is however inappropriate for cases which involve small bubbles that are of the size of the mesh element or smaller.

1.2 Numerical Solution for Free Surface Flows

The VOF system is the set of equations that govern the free surface flow and model the temporal and spatial changes of phases and their velocity, pressure and temperature distributions. The details of this system will be addressed in a later section. The interesting point to note here is that the different components of this system are strongly coupled to each other, so any numerical algorithm to solve it has to preserve this coupling. Another issue to point out here is the accuracy of the captured interface. Since the VOF method captures the interface algebraically, first order spatial discretization schemes would deteriorate the results. Therefore, compressive schemes or high order schemes have to be added to the method in order to accurately capture the interface. Turbulence handling is also an important point to raise when studying free surface flows due to the complications of the physics of the interface. The next sections will address these issues.

1.2.1 Velocity-Pressure-Volume Fraction Coupling

For free surface flows, the velocity and pressure fields are strongly affected by the position of the interface among the different phases, which in volume-of-fluid methods, is driven by the transport of the volume fraction fields. Therefore, the transport equations of volume fraction fields have to be coupled with the velocity-pressure system to produce robust solution. A system which has its velocity and pressure equations coupled together while the volume fraction transport equations are solved separately, is called segregated VOF system, while a system which has all the mentioned equations coupled together is called coupled VOF [11, 12]. More insight on the coupling strategy is provided in the literature review chapter.

1.2.2 Spatial and Temporal Discretization Schemes

Serious considerations should be made to generate an accurate as well as robust simulation of free surface flows. First, accurate temporal and spatial discretization schemes should be utilized in order to avoid any sort of numerical diffusion, which results in the attenuation of the accuracy of the interface. The interface between the phases cuts through computational cells because the volume fraction fields are averaged volume fractions within each cell. Thus, the interface sharpness has to be preserved, and this is attained by taking care of the discretization schemes, such as the spatial and transient schemes. Standard convective schemes are not suitable for advecting the volume fraction field as they do not preserve the interface. Various schemes have been used to advect the volume fraction field. High Resolution (HR) schemes have shown to be too diffusive, whereas compressive schemes have shown to be overly compressive, yielding a distorted interface, though sharp [13]. CICSAM and HRIC have been widely used also to capture the interface in free-surface flows [14, 15]. They yield an enhanced interface capturing, but they are very dependent on the Courant-number [16]. A new revolutionary interface capturing scheme denoted by STACS has been developed by [3], and has shown to have an improved performance as compared to standard and high order convective schemes, in that it is independent on the Courant-number. STACS is based on a switching strategy, by blending together the values of a compressive and a high-resolution advection scheme. The blending factor depends on the angle between the flow direction and the grid lines. A more ambitious method of compressing the interface, which strongly substitutes the compressive schemes mentioned earlier and is less sensitive to the Courant number, is the addition of the interfacial anti-diffusion term to the volume fraction equations, a method which is largely mentioned and manipulated in various forms and levels of treatment [17, 18, 19, 20, 21]. Finally, a compressive scheme which employs a limited gradient deferred correction approach exhibits a stable performance for high Courant numbers, with the limiter being designed according to Barth and Jesperson boundedness criteria [22] towards a downwind profile.

Regarding transient schemes in the VOF method, first-order Euler, second-order Euler, and Crank-Nicolson schemes have been widely adopted for transient-type free surface flows. However, the first order Euler scheme leads to undesirable numerical diffusion, while the second order upwind and Crank-Nicolson are highly dependent on the Courant-number. TICS transient discretization scheme, developed by [4], is a new methodology which switches between bounded high order and bounded compressive transient schemes. It has showed better interface sharpness as compared to a bounded second order upwind Euler scheme and Crank-Nicolson scheme.

1.2.3 Turbulence Handling

Almost all of the interface capturing methods are designed for laminar flows, and no considerations are given to turbulence conditions which are present in many complex applications of two-phase flows. Complex phenomena related to turbulence near the interface occur, many of which are still unresolved, or partially resolved. These involve mainly surface instabilities, transitional and turbulent regimes, and other physical processes/turbulence interactions such as moving boundaries, vortex interactions with the interface, rapid deformations of the interface, ligament formation, and breakup and merging of the interface. It can also be observed that the turbulence kinetic energy at the interface can be redistributed into the parallel surface components, leading to an anisotropic behavior [23, 24, 25]. Multiple approaches for simulating turbulence in such cases are present; Direct Numerical Simulation (DNS) [26], Large Eddy Simulation (LES) [27], Reynolds-Averaged-Navier-Stokes (RANS) [28, 29], to cite few. DNS and LES are too expensive and not appropriate for most industrial applications, while RANS models are widely used for simulating turbulence in industrial applications. However, despite its ability to model many engineering fluid flow cases, the two-equation eddy viscosity models that are based on RANS equations can fail in modeling violent distortions, or more generally non isotropic flows, in which the largest turbulent eddies can exhibit a complex behavior involving non-linear dependency on boundary conditions [30]. A more accurate approach, still based on RANS model, was first proposed by [31] for two-dimensional flows, then extended by [32] to three dimensions. We will refer to the formulation of this model as discussed by [33]. Referred to as "Explicit Algebraic Reynolds Stress Models (EARSM)", these closure forms consist of setting Reynolds stresses through an explicit dependency upon the rate-of-strain and vorticity tensors component. [34] has proved that EARSM provides better results as compared to a standard $k - \epsilon$ model in a shear flow within an open channel; in particular, the turbulent kinetic energy is improved. Numerical tests have shown that, in this particular case, the progress is mainly due to the corrected definition of production rate. As a consequence, eddy viscosity is also slightly better predicted. Solving for the turbulent Navier-Stokes equations, assuming eddy viscosity models, will require the addition of the turbulent viscosity to the laminar mixture viscosity in the stress term of the momentum equation. While for the EARSM model, the Reynolds stress tensor is explicitly calculated based on appropriate formulas and added as a source term in the momentum equation. The two issues are discussed and detailed in the appendix.

1.3 Outline of the Thesis

The current thesis addresses a novel coupled VOF methodology. A quick glance on the CFD science development over the past decades is stated in a literature review, chapter 2. Among the headlines listed in the chapter, a historical glance into the progress of CFD numerics is stated, followed by a presentation of the recent advances in the area of coupled VOF. Chapter 3 includes the definition of the VOF method from a physical and mathematical perspective. The chapter also discusses additional auxiliary models usually accompanying the VOF method. The finite volume discretization method for a general scalar equation and the VOF flow equations are addressed in details in chapter 4 along with other necessary features and techniques. Compressibility as well as buoyancy considerations are discussed in chapters 5 and 6 respectively. The linear solver theory which involves the iterative method for solving the coupled VOF algebraic system is presented in chapter 8. After that, the two chapters 9 and 10 present some assessment and performance evaluation for a set of free surface flow problems. A conclusion is then stated to summarize the whole theme as well as the outcomes. Appendices are also provided at the end of this thesis to discuss some selected topics relevant to the VOF method such as turbulence modeling, and other numerical and/or mathematical techniques used in the discretization process.

Chapter 2

Literature Review

2.1 Hierarchy of Fluid Flow Equations

The Navier-Stokes equations are the fundamental basis of almost all CFD problems. However, in the bigger picture, there's in fact a hierarchy of fluid flow equations which are usually solved with CFD. The conservation laws sit at the bottom of the fluid flow equations hierarchy, while other forms of the equations are derived from them depending on the physics of the problem. For example, the compressible Navier-Stokes equations are derived from continuum conservation laws, specifically, conservation of momentum in a fluid flow [35]. The Compressible Navier-Stokes equation is coupled to the mass conservation equation and also to the energy conservation equation. On another hand, the incompressible Navier-Stokes equation is the form derived from momentum conservation with the assumption that density is constant everywhere in the flow; it is still however coupled to the mass conservation, but now decoupled from the energy conservation equation [36]. For flows which involve negligible temperature variations, but significant buoyancy effects, the incompressible Navier-Stokes equation is still utilized but with a modified gravitational body force which is a function of the temperature; this is referred to as the Boussinesq approximation [37]. Thus, the energy conservation equation is again part of the solution. If multiple phases exist in the domain (aka multiphase flow), the same rules of physics hold except that each of the phases requires its own set of conservation laws [38]. A special case arises for the class of multiphase flows where normal and tangential stresses at the interface between the phases are negligible; this is called a free surface flow [39]. Then, all the phases would share the same set of conservation equations (mass, momentum and energy). In addition, multiphase flows involve phases that flow within the domain, a fact which requires an additional set of conservation equations which predict the location of the phases in space and time. The spatial and temporal variation of the location of each phase is described by a scalar conservation equation featuring an indicator quantity for the phase, usually selected to be the volume fraction of the phase [40, 41]. By all means, for any of the aforementioned fluid flow physics, a set of coupled and nonlinear partial differential equations exist. The unknowns are usually velocity, pressure, density, temperature, volume fraction, etc.

2.2 Computational Methods for Fluid Flow Problems

Solving the fluid flow equations using analytical tools is impossible, and scientists have largely referred to experiments to understand the behavior of fluid flow. For special types of problems, certain assumptions can be made, leading to simple and reduced order equations which can be solved analytically. A very popular example is the Couette flow, a viscous flow in laminar conditions moving between two surfaces. Real-life examples are flows in Earth's mantle and the atmosphere [42], flows constrained with journal bearings, viscometry and reversibility approximations [43]. When complex fluid flow problems are intended to be studied, and while experiments could be very costly and difficult to design and construct, computational techniques that approximate the solution are very possible and suitable. This branch of computational techniques is known as CFD, and it is nowadays an independent discipline in science [44, 45, 46]. CFD requires computers to perform the computations, and due to the exponential growth of computer power, larger and more complex problems are approached.

2.3 The Central Process in CFD

The central process of CFD is the discretization process of the fluid flow equations. The nonlinear partial differential equations are reduced to a linear system of algebraic equations, possible to solve within a computer. Different approaches exist for this purpose, such as the finite difference method, the finite element method, and the finite volume method. The finite volume method is the most incorporated discretization method for fluid flow problems, and it solves an integral form of the governing equations [47]. It is mentioned that the momentum conservation equation and the mass conservation equation are strongly coupled, and therefore another central dilemma here in CFD arises, that is, the way these strong couplings between the equations are resolved. A class of approaches referred to as the pressure-based approaches was originally developed in the 70's by Patankar [48]. His solution is devoted to low-speed incompressible flows, employed for staggered structural grids. This method has been extended, later in the 80's and 90's, to include compressible flows from low subsonic to high supersonic speeds [49, 50]. On the other hand, another class of approaches called density-based approaches were very popular in the domain of aeronautics and compressible flows [51, 52, 53, 54]. Density-based methods have also been extended to operate over low-speed flows with the use of some additional pre-conditioning techniques [55]. Using pressure-based or density-based methods, the velocity field is computed from the momentum equation. However, the pressure field in pressure-based methods is computed from the mass conservation equation after applying some reformulations and interpolations. In density-based methods, the pressure field is computed from the equation of state which contains a density field, computed prematurely from the mass conservation equation. Additional details of the methods can be found in [56, 57]. In the current PhD thesis, the pressure-based method is utilized for approaching the solution of the fluid flow system based on a finite volume method. Few notes on the pressure-based method are delivered here. The SIMPLE approach, abbreviation for semi-implicit method for pressure linked equations, is a widely used pressure-based numerical technique, developed by Spalding and his student Patankar [48, 58]; it is a simply implemented technique and requires low-memory. It has a segregated solution approach where equations are solved sequentially one after another. Other SIMPLE-like algorithms that were developed following the segregated approach also share similar behaviors [49, 59]. The drawback of this family of segregated solvers is the non-scalability, that is, they are very sensitive to the grid size. This is mainly due to the incorporation of under-relaxation, very necessary to maintain stable progress of the simulation through the iterations [60, 61]. The increasing knowledge of the CFD community has promoted the use of coupled solvers [62, 63, 64, 65], powered by possible access to huge computer memory in the recent decades. Unlike the segregated approach, the coupled approaches assemble all the governing equations in a single algebraic system. There have been lots of reports on the use of the coupled solvers for applications in incompressible and compressible flows [66, 67]. The coupling has basically targeted momentum and mass conservation equations due to the strong coupling between velocity and pressure fields. However, additional coupling strategies have also been an active area of research, such as energy conservation equation coupling with both momentum and mass conservation equations for compressible flows [68], and multiphase flow coupling [69, 70, 61]. This PhD thesis sits at this exact place in this research area, which is, coupling the conservation equations in the field of free surface flow solution as much as possible to render full-implicit systems, aiming at enhancing the robustness of the solution. As reported previously, the VOF method is used in this context to simulate free surface flows. Momentum and mass conservation equations are to be coupled with volume fraction equations to produce a fully coupled VOF system. The following section addresses the recent advancement in the domain of coupled VOF.

2.4 Recent Advances in VOF Coupled Approaches

Previous efforts for solving the VOF system by means of coupled approaches are reported here. Denner et al. [71] have shown in their work a coupled VOF for two-phase flows where they do one-way coupling between momentum and pressure equations with volume fraction field through the surface tension force term. In their paper, Zwart et al. [11] have developed a fully coupled VOF solver where they couple the momentum equation to the volume fraction equations through the fully implicit body force term, and they also couple the volume fraction equations to the momentum and pressure equations through the use of linearization techniques. However, they lack any sort of comparison for the effectiveness of their solver with the conventional VOF solvers. Also, [12] have developed a coupled VOF solver for a two-phase flow proving a superiority of this solver over the standard segregated VOF one, but they only mention a coupling of the momentum equation with the volume fraction field through the body forces, in addition to coupling the volume fraction equation to both velocity and pressure fields. In [72], they develop a pressure-based VOF coupled method for two or more immiscible fluids through which they couple the pressure equation with the volume fraction fields, and in turn, they couple the volume fraction equations with the pressure field. Their coupling has the advantage of preventing the numerically induced predominance of a single phase which may take place usually as a result of a segregated procedure of the solution of VOF method. To this end, this area of research is still largely active, yet very challenging in terms of math and numerics, nevertheless, great efforts are being put to turn fruits out of the coupling approaches of the VOF method. This thesis attempts to uncover a proper methodology with sufficient-level of details for coupling the free surface flow equations in a single system, which is to be solved simultaneously.

Chapter 3

Theoretical Background

3.1 The VOF Method

In this section the mathematical representation for the physics of free surface flows based on the VOF method is presented. In VOF, all the phases are assumed immiscible and homogeneous, and they share the pressure, velocity, temperature and turbulence fields. Thus, a single set of flow equations is solved with varying transport and thermophysical properties (density, viscosity, thermal conductivity, etc.); this variation is determined by the location of the phases in the domain. Assuming a laminar flow with all phases being incompressible, the mathematical representation of the Navier-Stokes and continuity equations is given in equations 3.1 and 3.2 by

$$\frac{\partial \rho \mathbf{v}}{\partial t} + \nabla \cdot (\rho \mathbf{v}\mathbf{v}) = \nabla \cdot \tau - \nabla p + \mathbf{B} \tag{3.1}$$

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \tag{3.2}$$

τ in the Navier-Stokes equation is the stress tensor defined as:

$$\tau = \mu \left(\nabla \mathbf{v} + \nabla \mathbf{v}^T \right) \tag{3.3}$$

$\mathbf{B}$ in the momentum conservation equation is a body force term per unit volume, such as gravitational force $\mathbf{f}_g = \rho \mathbf{g}$ and/or surface tension force $\mathbf{f}_s$ to be addressed in the next section. ρ and μ refer to density and viscosity, and they are expressed as volumetric mixture properties in a flow which consists of n fluids. These properties can be directly related to the properties of each phase k by its volume fraction α^k such that

$$\rho = \sum_{k=1}^{n} \alpha^k \rho^k \tag{3.4}$$

and

$$\mu = \sum_{k=1}^{n} \alpha^k \mu^k \tag{3.5}$$

α^k is also called an indicator function, and is computed by solving a scalar convection equation defined according to Hirt and Nichols [8] by:

$$\frac{\partial \rho^k \alpha^k}{\partial t} + \nabla \cdot (\rho^k \mathbf{v} \alpha^k) = 0 \tag{3.6}$$

3.2 Geometric Conservation of Volume Fraction Fields

The volume fraction fields must sum to one. This is called geometric conservation of volume fraction fields, and it is a constraint which has to be maintained at all circumstances. It is formulated as

$$\sum_{k=1}^{n} \alpha^k = 1 \tag{3.7}$$

In order to maintain the constraint eq. 3.7, different approaches have been reported to solve the volume fraction equations in the VOF method. The conventional approach solves $n-1$ volume fraction equations corresponding to $n-1$ phases in the flow, and then it applies the geometric conservation constraint to determine the n^{th} phase volume fraction (this phase is referred to as predominant phase). There two major drawbacks of such an approach. First, numerical errors arising from the solution of the $n-1$ equations are inherited by the n^{th} volume fraction fields. Second, though this approach mathematically preserves geometric conservation, it cannot be set as a generalized framework for solving multiphase problems. For example, the fact that volume fraction equations of low-density materials have poor convergence characteristics, the volume fraction field of the lightest material must be selected as n^{th} field to be determined explicitly. The other approach solves n volume fraction equations to ensure that errors of the solution are not transferred to the explicitly determined field, while it is also a general approach with no regard to density rankings. This approach has been referred to in Moukalled et. al [73]. However, the approach does not guarantee a bounded solution, due to numerical difficulties. It also does not guarantee the geometric conservation principle unless the constraint is embedded in the solution. Embedding the constraint could be done as an explicit normalization step, either before the assembly of the conservation equations or after the solution of the volume fraction equations. However, this step would deteriorate the convergence rate of volume fraction equations, especially when mass transfer phenomena are involved. An implicit normalization method addressed by [74] is another method which incorporates the geometric constraint within the volume fraction equations. This implementation leads to geometrically conserved volume fraction fields without altering the volume fraction fields in an extra step. The method is explained in the discretization section of the volume fraction equations.

3.3 Surface Tension and Wall Adhesion

3.3.1 Continuum Surface Force Model (CSF)

In this work we adopt the continuum surface force formulation (CSF) for calculating the surface tension force $\mathbf{f}_s$ as presented by Brackbill et al. [75]. The force at the surface can be expressed as a volume force using the divergence theorem. The volumetric surface force has the following form:

$$\mathbf{f}_s = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \sigma^{ij} \frac{\alpha^i \rho^i \kappa^j \nabla \alpha^j + \alpha^j \rho^j \kappa^i \nabla \alpha^i}{\frac{1}{2}\left(\rho^i + \rho^j\right)} \tag{3.8}$$

where σ^{ij} is the surface tension coefficient between a pair of fluids i and j. κ^i is the curvature of the interface, defined in terms of the unit normal vector $\hat{\mathbf{n}}$ of the curvature outward of the fluid i region such that:

$$\kappa^i = \nabla \cdot \hat{\mathbf{n}} \tag{3.9}$$

and

$$\hat{\mathbf{n}} = \frac{\nabla \alpha^i}{|\nabla \alpha^i|} \tag{3.10}$$

3.3.2 Wall Adhesion

The wall adhesion angle according to the work of Brackbill et al. [75], adjusts the surface normal in the boundary element adjacent to the wall. Denoting the contact angle at the wall by θ_w, the surface normal at the adjacent element near the wall is:

$$\hat{n} = \hat{n}_w \cos\theta_w + \hat{t}_w \sin\theta_w \tag{3.11}$$

with $\hat{n}_w$ and $\hat{t}_w$ being respectively the unit normal and tangential vectors to the wall. Accounting for this adjustment in the surface normal vector leads to an updated free surface curvature locally at the wall, which in turn influences the calculation of the surface tension force term.

3.4 Interfacial Anti-Diffusion Term

Referring back to the phasic equation (eq. 3.6), high resolution advection schemes must be utilized to preserve a sharp interface between the phases. Among other algebraic methods, a method which has recently gained popularity is the addition of an interfacial anti-diffusion term, called also compression term, which is added to the scalar equation of α^k, and is capable of compressing the surface towards a sharper one [76]. The phasic equation will be as follows:

$$\frac{\partial \rho^k \alpha^k}{\partial t} + \nabla \cdot (\rho^k \mathbf{v} \alpha^k) + \nabla \cdot \left[\rho^k \alpha^k (1 - \alpha^k) \mathbf{v}_r\right] = 0 \tag{3.12}$$

where $\mathbf{v}_r$ is the relative velocity or the *compression* velocity. This term avoids the use of high resolution compressive schemes such as CICSAM for instance [77]. Further details related to the anti-diffusion term will be discussed in the discretization section.

3.5 Pressure Shift for Incompressible Flows

The pressure field in the Navier-Stokes equation 3.1 appears within a gradient operator ∇, a fact which states that only the change of pressure is what drives the flow. This is not to be confused with the necessity of the pressure field itself in evaluating different thermodynamic properties such as cases which include compressibility effects. The current note on the pressure shift applies only for flows of incompressible phases. Following are two types of pressure shifts, the first is due to a fixed operating pressure and the other due to a reference density.

3.5.1 Pressure Shift due to a Fixed Operating Pressure

For incompressible flows, it is advisable to incorporate a shifted pressure instead of the real pressure; this is attained by removing a reference pressure level. The shifted pressure in this sense is said to be a gauge pressure which is measured above some reference pressure at which the problem is operating. Commonly, this operating reference pressure is the atmospheric pressure, i.e. 101325 pa at sea level. This shift in pressure prevents the drawback of round-off errors in the arithmetic operations, especially for problems which exhibit small pressure variations. Therefore, the real pressure p can be decomposed into a reference pressure (i.e. operating pressure) and a relative pressure (i.e. gauge pressure):

$$p = p_{rel} + p_{ref} \tag{3.13}$$

applying gradient operator to both sides of the previous equation:

$$\nabla p = \nabla p_{rel} + \nabla p_{ref} \tag{3.14}$$

p_{ref} is a fixed pressure level, so it is zero under the gradient operator, then:

$$\nabla p = \nabla p_{rel} \tag{3.15}$$

Eq. 3.15 ensures that the use of a relative pressure will lead to the same driving force mathematically, but as mentioned before, from a numerical point of view, its use is very essential to avoid round-off errors.

3.5.2 Pressure Shift due to Reference Density

On another hand, the real pressure p in a flow where gravity $\mathbf{g}$ is active, can also be decomposed as such:

$$p = p_{rel} + \rho_0 \mathbf{g} \cdot (\mathbf{R} - \mathbf{R}_{ref}) \tag{3.16}$$

where p_{rel} is the relative (shifted) pressure, while ρ_0 is a reference density usually taken to be the density of the lightest fluid in the domain, and $\mathbf{R}_{ref}$ is a reference location vector, usually taken to be the centroid of a pressure boundary, or simply the pressure reference location. The body force term in the Navier-Stokes equation can be re-formulated into a more appropriate form, by considering equation 3.16. Applying a gradient operator on both sides of equation 3.16:

$$\nabla p = \nabla p_{rel} + \nabla \left[\rho_0 \mathbf{g} \cdot (\mathbf{R} - \mathbf{R}_{ref})\right] = \nabla p_{rel} + \rho_0 \mathbf{g} \tag{3.17}$$

Then, substituting ∇p in the Navier-Stokes equation 3.1, and replacing the body force $\mathbf{B}$ by the gravitational force $\rho \mathbf{g}$, leads to the following form of the equation:

$$\frac{\partial \rho \mathbf{v}}{\partial t} + \nabla \cdot (\rho \mathbf{v}\mathbf{v}) = \nabla \cdot \tau - \nabla p_{rel} + (\rho - \rho_0)\mathbf{g} + \mathbf{f}_s \tag{3.18}$$

This form of pressure will be used throughout the report, and for simplicity, the *rel* subscript will be removed, so the shifted pressure due to reference density and due to a fixed operating pressure as well, is p hereafter.

Chapter 4

Finite Volume Discretization

4.1 Finite Volume Mesh

The discretization of the mathematical models is based on a cell-centered finite volume method. The current discretization assumes the most general form of a cell-centered grid, which is an unstructured non-orthogonal grid, with no element-type constraint. Figure 4.1 shows an unstructured cell-centered finite volume mesh. There are important geometric features of the mesh element/face that go directly into the discretization process. Figure 4.2 summarizes the important geometric quantitative features. Also, some qualitative features of the face in an unstructured finite volume mesh are non-orthogonality and skewness. These

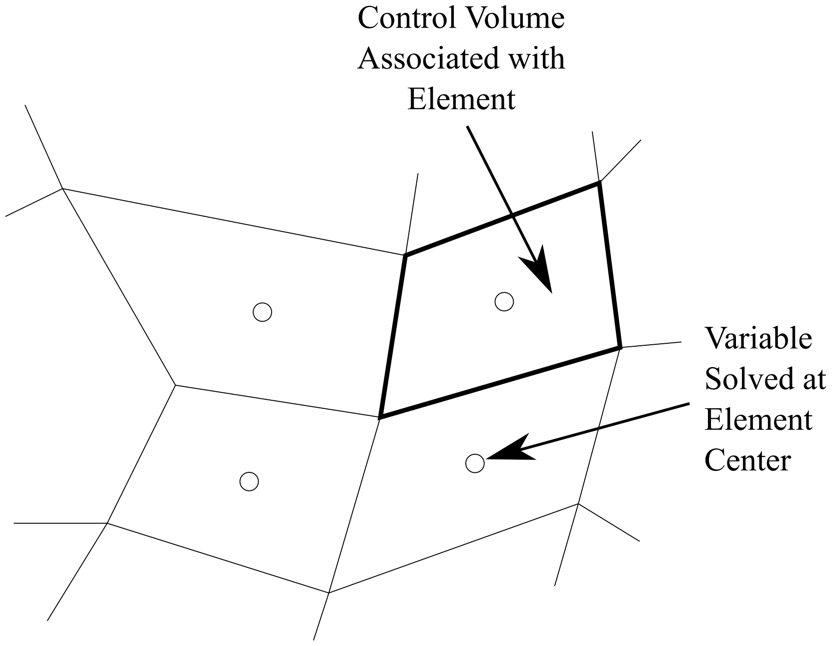


Figure 4.1: Unstructured finite volume cell-centered mesh

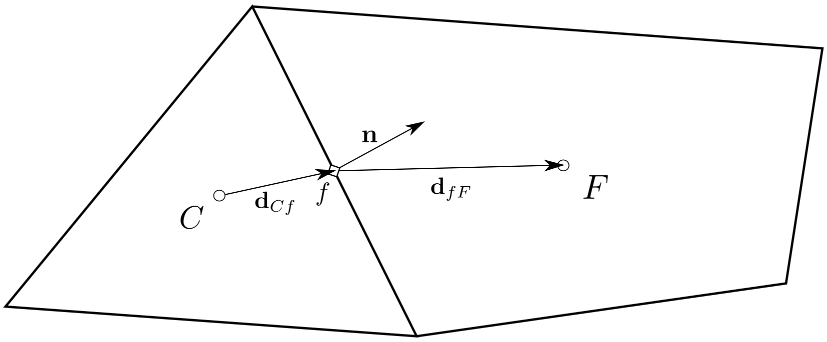
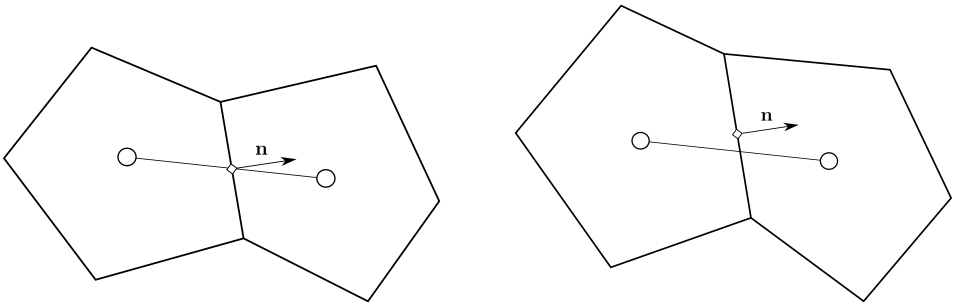


Figure 4.2: Geometric features



(a) Face non-orthogonality

(b) Face skewness

Figure 4.3: Geometric features of an unstructured finite volume mesh

features are critical to the discretization process. This is shown in figure 4.3. Face non-orthogonality (figure 4.3a) is the non-alignment of the face normal with the line connecting the neighboring cells' centers. This is a geometric feature usually sensitive to gradient computation at the face, and it has a set of treatments to avoid its consequences. Face skewness (figure 4.3b) is the non-coincidence of the intersection point of the line connecting two neighboring cell centers with the face at its center. This geometric feature has usually a limited consequence, and may only shortly affect the accuracy of the results. $\mathbf{n}$ in the figure is the face normal unit vector. Figure 4.4 shows the surface vector $\mathbf{S}_f$ of the face f between neighbour elements C and F, which can be decomposed into two components $\mathbf{E}_f$ and $\mathbf{T}_f$, such that:

$$\mathbf{S}_f = \mathbf{E}_f + \mathbf{T}_f \tag{4.1}$$

where $\mathbf{E}_f$ is in the direction of $\mathbf{CF}$ and has the same norm as that of $\mathbf{S}_f$, while $\mathbf{T}_f$ is the complement of $\mathbf{E}_f$.

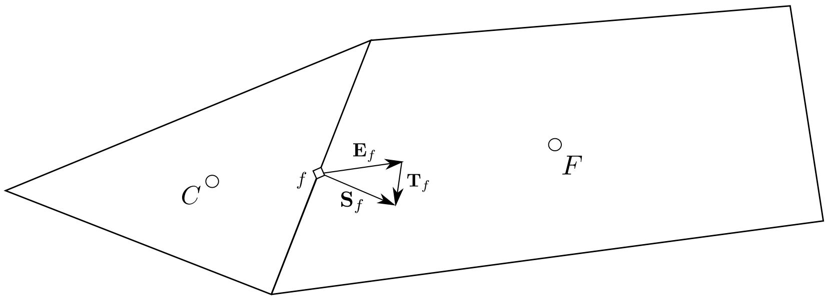


Figure 4.4: Neighbouring elements C and F with face f between them, showing the decomposition of the surface vector $\mathbf{S}_f$ into its ortho/non-ortho components

4.2 General Scalar Equation

4.2.1 Preliminaries

The discretization procedures will adopt the superscripts Asterisk * and Degree $^\circ$ to refer to the latest value and previous time step value respectively. Also, the bar over any quantity $\overline{\phi}_f$ at a face is an interpolated quantity from the neighboring cells to the face such that, assuming a linear interpolation:

$$\overline{\phi}_f = g_f \phi_C + (1 - g_f)\phi_F \tag{4.2}$$

where C and F are two neighbouring elements as can be shown in figure 4.4, and g_f is a geometric weighting factor given by:

$$g_f = \frac{V_F}{V_C + V_F} \tag{4.3}$$

g_f may also be defined in terms of geometric vectors such that:

$$g_f = \frac{-\mathbf{d}_{Ff} \cdot \mathbf{n}}{-\mathbf{d}_{Ff} \cdot \mathbf{n} + \mathbf{d}_{Cf} \cdot \mathbf{n}} \tag{4.4}$$

Other interpolation schemes will be addressed accordingly. The $||a, b||$ notation will be used to represent a maximum between two quantities a and b such that:

$$||a, b|| = \max(a, b) \tag{4.5}$$

4.2.2 Discretization

The general conservation equation of a scalar field ϕ has the following form:

$$\frac{\partial \rho \phi}{\partial t} + \nabla \cdot (\rho \mathbf{v} \phi) = \nabla \cdot (\Gamma^\phi \nabla \phi) + Q^\phi \tag{4.6}$$

Integrating the above equation over a control volume:

$$\int \frac{\partial \rho \phi}{\partial t} dV + \int \nabla \cdot (\rho \mathbf{v} \phi) dV = \int \nabla \cdot (\Gamma^{\phi} \nabla \phi) dV + \int Q^{\phi} dV \tag{4.7}$$

Applying the finite volume principles, including divergence theorem and midpoint integration rule, and others, the semi-discrete form of the equation over a finite volume C is:

$$\frac{(\rho_C \phi_C) - (\rho_C \phi_C)^{\circ}}{\Delta t} V_C + \sum_{f \sim nb(C)} \dot{m}_f \phi_f = \sum_{f \sim nb(C)} \Gamma_f^{\phi} \nabla \phi_f \cdot \mathbf{S}_f + Q_C^{\phi} V_C \tag{4.8}$$

Assuming the following schemes for ϕ: first order Euler transient scheme, first order upwind convection scheme, and central difference scheme for diffusion, and considering an unstructured non-orthogonal mesh, the corresponding discrete algebraic equation over the element C is:

$$a_C \phi_C + \sum_{F \sim NB(C)} a_F \phi_F = b_C \tag{4.9}$$

where

$$\begin{aligned}
a_C &= \frac{\rho_C^*}{\Delta t} V_C + \sum_{f \sim nb(C)} \left(||\dot{m}_f^*, 0|| + \Gamma_f^{\phi,*} \frac{E_f}{d_{CF}} \right) \\
a_F &= -|| - \dot{m}_f^*, 0|| - \Gamma_f^{\phi,*} \frac{E_f}{d_{CF}} \\
b_C &= \frac{(\rho_C \phi_C)^{\circ}}{\Delta t} V_C + \sum_{f \sim nb(C)} \Gamma_f^{\phi,*} \nabla \phi_f^* \cdot \mathbf{T}_f + Q_C^{\phi} V_C
\end{aligned}$$

The boundary conditions for ϕ could either be a specified value of ϕ, called ϕ_b, or a specified flux q_b through the boundary face b. For inlet or outlet patches, if ϕ is specified, the coefficients of eq. 4.8 is modified as follows:

$$a_C = a_C + \Gamma_b^* \frac{E_b}{d_{Cb}} \tag{4.10}$$

$$b_C = b_C - \dot{m}_b^* \phi_b + \Gamma_b^* \frac{E_b}{d_{Cb}} \phi_b + \Gamma_b^* \nabla \phi_b \cdot \mathbf{T}_b \tag{4.11}$$

However, if the considered physical patch is either a wall or a symmetry plane, the coefficients would be:

$$a_C = a_C + \Gamma_b^* \frac{E_b}{d_{Cb}} \tag{4.12}$$

$$b_C = b_C \Gamma_b^* \frac{E_b}{d_{Cb}} \phi_b + \Gamma_b^* \nabla \phi_b \cdot \mathbf{T}_b \tag{4.13}$$

In the other case, where a flux q_b outward of the boundary is specified, it is added to the right-hand-side of the equation in case of a wall or a symmetry plane such that:

$$b_C = b_C - q_b S_b \tag{4.14}$$

while for an inlet or outlet, the flux q_b is usually zero for most of the problems, and the value of ϕ at the boundary is extrapolated from the interior such that $\phi_b = \phi_C$, so it alters the coefficients by:

$$a_C = a_C + ||\dot{m}_b^*, 0|| \tag{4.15}$$

$$b_C = b_C + || - \dot{m}_b^*, 0||\phi_b^* \tag{4.16}$$

4.3 Segregated VOF System

As mentioned earlier, the conventional procedure for solving the VOF system is by solving for velocity and pressure first, and then solving the volume fraction equations ahead of that. The velocity and pressure fields may be solved simultaneously, and this is usually referred to as velocity-pressure coupled approach. The current discretization methodology considers an incompressible laminar flow, retrieved from ref. [62]. The discrete algebraic system of the velocity-pressure coupled system for an element C in the computational domain is of the form:

$$\begin{bmatrix} a_C^{uu} & a_C^{uv} & a_C^{uw} & a_C^{up} \\ a_C^{vu} & a_C^{vv} & a_C^{vw} & a_C^{vp} \\ a_C^{wu} & a_C^{wv} & a_C^{ww} & a_C^{wp} \\ a_C^{pu} & a_C^{pv} & a_C^{pw} & a_C^{pp} \end{bmatrix} \begin{bmatrix} u_C \\ v_C \\ w_C \\ p_C \end{bmatrix} + \sum_{F \sim NB(C)} \begin{bmatrix} a_F^{uu} & a_F^{uv} & a_F^{uw} & a_F^{up} \\ a_F^{vu} & a_F^{vv} & a_F^{vw} & a_F^{vp} \\ a_F^{wu} & a_F^{wv} & a_F^{ww} & a_F^{wp} \\ a_F^{pu} & a_F^{pv} & a_F^{pw} & a_F^{pp} \end{bmatrix} \begin{bmatrix} u_F \\ v_F \\ w_F \\ p_F \end{bmatrix} = \begin{bmatrix} b_C^u \\ b_C^v \\ b_C^w \\ b_C^p \end{bmatrix}$$

The volume fraction equations are n non-coupled equations, solved individually and sequentially, $\forall\ k \in [1, n]$ as shown here:

$$a_C^{\alpha^k} \alpha_C^k + \sum_{F \sim NB(C)} a_F^{\alpha^k} \alpha_F^k = b_C^{\alpha^k}$$

4.3.1 Momentum Conservation Equation

Following the same discritization principles presented earlier for a general scalar equation, the semi-discrete form of the momentum equation (eq. 3.1) over an element C is:

$$\begin{aligned} &\frac{(\rho_C \mathbf{v}_C) - (\rho_C \mathbf{v}_C)^\circ}{\Delta t} V_C + \sum_{f \sim nb(C)} \dot{m}_f \mathbf{v}_f \\ &= \sum_{f \sim nb(C)} \mu_f \left(\nabla \mathbf{v}_f + \nabla \mathbf{v}_f^T \right) \cdot \mathbf{S}_f - \sum_{f \sim nb(C)} p_f \mathbf{S}_f + \mathbf{B}_C V_C \end{aligned} \tag{4.17}$$

Without any further details, the algebraic equation of the previous semi-discretized momentum equation for the three components:

$$\begin{aligned} &a_C^{uu}u_C + \underline{a_C^{uv}v_C} + \underline{a_C^{uw}w_C} + a_C^{up}p_C \\ &\qquad + \sum_{F\sim NB(C)} a_F^{uu}u_F + \underline{\sum_{F\sim NB(C)} a_F^{uv}v_F} + \underline{\sum_{F\sim NB(C)} a_F^{uw}w_F} + \sum_{F\sim NB(C)} a_F^{up}p_F = b_C^u \end{aligned}$$

$$\begin{aligned} &\underline{a_C^{vu}u_C} + a_C^{vv}v_C + \underline{a_C^{vw}w_C} + a_C^{vp}p_C \\ &\qquad + \underline{\sum_{F\sim NB(C)} a_F^{vu}u_F} + \sum_{F\sim NB(C)} a_F^{vv}v_F + \underline{\sum_{F\sim NB(C)} a_F^{vw}w_F} + \sum_{F\sim NB(C)} a_F^{vp}p_F = b_C^v \end{aligned}$$

$$\begin{aligned} &\underline{a_C^{wu}u_C} + \underline{a_C^{wv}v_C} + a_C^{ww}w_C + a_C^{wp}p_C \\ &\qquad + \underline{\sum_{F\sim NB(C)} a_F^{wu}u_F} + \underline{\sum_{F\sim NB(C)} a_F^{wv}v_F} + \sum_{F\sim NB(C)} a_F^{ww}w_F + \sum_{F\sim NB(C)} a_F^{wp}p_F = b_C^w \end{aligned}$$

where

$$\begin{aligned} a_C^{uu} = a_C^{vv} = a_C^{ww} &= \frac{\rho_C^*}{\Delta t}V_C + \sum_{f\sim nb(C)} \left(||\dot{m}_f^*, 0|| + \mu_f^* \frac{E_f}{d_{CF}} \right) \\ a_F^{uu} = a_F^{vv} = a_F^{ww} &= -|| - \dot{m}_f^*, 0|| - \mu_f^* \frac{E_f}{d_{CF}} \end{aligned}$$

$$\begin{aligned} a_C^{up} &= \sum_{f\sim nb(C)} g_f S_f^x & a_C^{vp} &= \sum_{f\sim nb(C)} g_f S_f^y & a_C^{wp} &= \sum_{f\sim nb(C)} g_f S_f^z \\ a_F^{up} &= (1-g_f)S_f^x & a_F^{vp} &= (1-g_f)S_f^y & a_F^{wp} &= (1-g_f)S_f^z \end{aligned}$$

$$\begin{aligned} b_C^u &= \sum_{f\sim nb(C)} [\nabla u^* \cdot \mathbf{T}_f] + \frac{(\rho_C u_C)^\circ}{\Delta t}V_C + \sum_{f\sim nb(C)} \left[(\mu_f \nabla \mathbf{v}_f^T \cdot \mathbf{S}_f)^* \cdot \hat{\mathbf{i}} \right] + \mathbf{B}_C^x V_C \\ b_C^v &= \sum_{f\sim nb(C)} [\nabla v^* \cdot \mathbf{T}_f] + \frac{(\rho_C v_C)^\circ}{\Delta t}V_C + \sum_{f\sim nb(C)} \left[(\mu_f \nabla \mathbf{v}_f^T \cdot \mathbf{S}_f)^* \cdot \hat{\mathbf{j}} \right] + \mathbf{B}_C^y V_C \\ b_C^w &= \sum_{f\sim nb(C)} [\nabla w^* \cdot \mathbf{T}_f] + \frac{(\rho_C w_C)^\circ}{\Delta t}V_C + \sum_{f\sim nb(C)} \left[(\mu_f \nabla \mathbf{v}_f^T \cdot \mathbf{S}_f)^* \cdot \hat{\mathbf{k}} \right] + \mathbf{B}_C^z V_C \end{aligned}$$

It should be noted that a first order Euler transient scheme and a first order upwind convection scheme for the transient and convection terms respectively are assumed. Moreover, the single underlined terms in the algebraic equation account for the velocity component interactions with their values being zero except at wall boundaries and symmetry planes.

4.3.2 Mass Conservation Equation

For practical reasons, the continuity equation (eq. 3.2) which is a summation of all phasic equations can be expressed in a scaled form in order to avoid round-off errors. The following equation is basically the continuity equation in its phasic build-up form:

$$\sum_{k=1}^{n}\left[\frac{\partial \rho^k \alpha^k}{\partial t} + \nabla \cdot \left(\rho^k \mathbf{v} \alpha^k\right) = 0\right] \tag{4.18}$$

Now this equation can be scaled by dividing by some quantity, such as a density scale:

$$\sum_{k=1}^{n}\left[\frac{1}{\rho^k_{scale}}\frac{\partial \rho^k \alpha^k}{\partial t} + \frac{1}{\rho^k_{scale}}\nabla \cdot \left(\rho^k \mathbf{v} \alpha^k\right) = 0\right] \tag{4.19}$$

Recalling that the fluids in this analysis are considered incompressible, so ρ^k is constant, and realizing that $\rho^k_{scale} = \rho^k = cte$, the equation above can be re-written as follows:

$$\sum_{k=1}^{n}\left[\frac{\partial \alpha^k}{\partial t} + \frac{1}{\rho^k_{scale}}\nabla \cdot \left(\rho^k \mathbf{v} \alpha^k\right) = 0\right] \tag{4.20}$$

The scaling in the other equation is kept without any further simplification in order to reach a form where a phasic mass flow rate $\dot{m}^k_f$ can be used. However, the first term, $\frac{\partial \alpha^k}{\partial t}$, can be dropped out because the cumulative time rate of change is zero:

$$\sum_{k=1}^{n}\frac{\partial \alpha^k}{\partial t} = \frac{\partial}{\partial t}\sum_{k=1}^{n}\alpha^k = \frac{\partial}{\partial t}(1) = 0 \tag{4.21}$$

Therefore, the final version of the continuity equation which will be utilized for incompressible flows is:

$$\sum_{k=1}^{n}\left[\frac{1}{\rho^k_{scale}}\nabla \cdot \left(\rho^k \mathbf{v} \alpha^k\right) = 0\right] \tag{4.22}$$

The semi-discrete form of the pressure equation expressed in its latest form at an arbitrary volume C is written as follows:

$$\sum_{k=1}^{n}\left[\frac{1}{\rho^k_{scale}}\sum_{f\sim nb(C)}\alpha^k_f \rho^k_f \mathbf{v}_f \cdot \mathbf{S}_f = 0\right] \tag{4.23}$$

The Rhie-Chow interpolation [78] of $\mathbf{v}$ at a face f is:

$$\mathbf{v}_f = \overline{\mathbf{v}_f} - \overline{\mathbf{D}^{\mathbf{v}}_f}\left(\nabla p_f - \overline{\nabla p_f}\right) \tag{4.24}$$

∇p_f is the small-stencil face-gradient of pressure, calculated based on a corrected face-gradient formula:

$$\nabla p_f = \overline{\nabla p_f} - \left(\overline{\nabla p_f} \cdot \mathbf{e}_f - \frac{p_F - p_C}{d_{CF}} \right) \mathbf{e}_f \tag{4.25}$$

The large-stencil face pressure gradient $\overline{\nabla p_f}$ is determined using a linear interpolation for the straddling elements' pressure gradients:

$$\overline{\nabla p_f} = g_f \nabla p_C + (1 - g_f) \nabla p_F \tag{4.26}$$

and the pressure gradient at an arbitrary element C is calculated using Gauss formula:

$$\nabla p_C = \frac{1}{V_C} \sum_{f \sim nb(C)} p_f \mathbf{S}_f \tag{4.27}$$

The face pressure p_f is linearly interpolated:

$$p_f = g_f p_C + (1 - g_f) p_F \tag{4.28}$$

$\overline{\mathbf{D}_f^{\mathbf{v}}}$ is an adaptive tensor linearly interpolated at a face from the neighbouring cell values:

$$\overline{\mathbf{D}_f^{\mathbf{v}}} = g_f \mathbf{D}_C^{\mathbf{v}} + (1 - g_f) \mathbf{D}_F^{\mathbf{v}} \tag{4.29}$$

and for an element C, $\mathbf{D}_C^{\mathbf{v}}$ can be expressed such as:

$$\mathbf{D}_C^{\mathbf{v}} = \begin{bmatrix} D_C^u & 0 & 0 \\ 0 & D_C^v & 0 \\ 0 & 0 & D_C^w \end{bmatrix} = \begin{bmatrix} \frac{V_C}{a_C^{uu}} & 0 & 0 \\ 0 & \frac{V_C}{a_C^{vv}} & 0 \\ 0 & 0 & \frac{V_C}{a_C^{ww}} \end{bmatrix} \tag{4.30}$$

Substituting in eq. 4.23, the semi-discrete equation becomes:

$$\sum_{k=1}^{n} \left\{ \frac{1}{\rho_{scale}^k} \sum_{f \sim nb(C)} \alpha_f^k \rho_f^k \left[\overline{\mathbf{v}_f} - \overline{\mathbf{D}_f^{\mathbf{v}}} \left(\nabla p_f - \overline{\nabla p_f} \right) \right] \cdot \mathbf{S}_f = 0 \right\} \tag{4.31}$$

The semi-discrete pressure equation can be expanded:

$$\begin{aligned} \sum_{k=1}^{n} \Bigg[& \frac{1}{\rho_{scale}^k} \sum_{f \sim nb(C)} \alpha_f^k \rho_f^k (-\overline{\mathbf{D}_f^{\mathbf{v}}} \nabla p_f) \cdot \mathbf{S}_f + \frac{1}{\rho_{scale}^k} \sum_{f \sim nb(C)} \alpha_f^k \rho_f^k \overline{\mathbf{v}_f} \cdot \mathbf{S}_f \\ & = \frac{1}{\rho_{scale}^k} \sum_{f \sim nb(C)} \alpha_f^k \rho_f^k (-\overline{\mathbf{D}_f^{\mathbf{v}}}\ \overline{\nabla p_f}) \cdot \mathbf{S}_f \Bigg] \end{aligned} \tag{4.32}$$

The average velocity at the face is expressed as:

$$\overline{\mathbf{v}_f} = g_f \mathbf{v}_C + (1 - g_f) \mathbf{v}_F \tag{4.33}$$

The algebraic equations for the discrete continuity equation:

$$\begin{aligned}
&a_C^{pu} u_C + a_C^{pv} v_C + a_C^{pw} w_C + a_C^{pp} p_C \\
&+ \sum_{F\sim NB(C)} a_F^{pu} u_F + \sum_{F\sim NB(C)} a_F^{pv} v_F + \sum_{F\sim NB(C)} a_F^{pw} w_F + \sum_{F\sim NB(C)} a_F^{pp} p_F = b_C^p
\end{aligned}$$

where

$$\begin{aligned}
a_C^{pu} &= \sum_{k=1}^{n} \left[\frac{1}{\rho_{scale}^k} \sum_{f\sim nb(C)} (\alpha_f^k \rho_f^k)^* g_f S_f^x \right] \\
a_C^{pv} &= \sum_{k=1}^{n} \left[\frac{1}{\rho_{scale}^k} \sum_{f\sim nb(C)} (\alpha_f^k \rho_f^k)^* g_f S_f^y \right] \\
a_C^{pw} &= \sum_{k=1}^{n} \left[\frac{1}{\rho_{scale}^k} \sum_{f\sim nb(C)} (\alpha_f^k \rho_f^k)^* g_f S_f^z \right]
\end{aligned}$$

$$a_C^{pp} = \sum_{k=1}^{n} \left[\frac{1}{\rho_{scale}^k} \sum_{f\sim nb(C)} (\alpha_f^k \rho_f^k)^* \mathcal{D}_f \right]$$

$$\begin{aligned}
a_F^{pu} &= \sum_{k=1}^{n} \left[\frac{1}{\rho_{scale}^k} (\alpha_f^k \rho_f^k)^* (1 - g_f) S_f^x \right] \\
a_F^{pv} &= \sum_{k=1}^{n} \left[\frac{1}{\rho_{scale}^k} (\alpha_f^k \rho_f^k)^* (1 - g_f) S_f^y \right] \\
a_F^{pw} &= \sum_{k=1}^{n} \left[\frac{1}{\rho_{scale}^k} (\alpha_f^k \rho_f^k)^* (1 - g_f) S_f^z \right]
\end{aligned}$$

$$a_F^{pp} = -\sum_{k=1}^{n} \left[\frac{1}{\rho_{scale}^k} (\alpha_f^k \rho_f^k)^* \mathcal{D}_f \right]$$

$$b_C^p = \sum_{k=1}^{n} \left[\frac{1}{\rho_{scale}^k} \sum_{f\sim nb(C)} \left(\alpha_f^k \rho_f^k\right)^* (\overline{\mathbf{D}_f^{\mathbf{v}}}\ \overline{\nabla p_f^*}) \cdot \mathbf{S}_f + \frac{1}{\rho_{scale}^k} \sum_{f\sim nb(C)} \left(\alpha_f^k \rho_f^k\right)^* (\overline{\mathbf{D}_f^{\mathbf{v}}}\ \nabla p_f^*) \cdot \mathbf{T}_f \right]$$

and

$$\mathcal{D}_f = \frac{\sqrt{(\overline{D_f^u} S_f^x)^2 + (\overline{D_f^v} S_f^y)^2 + (\overline{D_f^w} S_f^z)^2}}{d_{CF}}$$

4.3.3 Volume Fraction Equation

The volume fraction equation for a phase k presented previously in equation 3.6 is presented here in semi-discrete form:

$$\frac{(\rho_C^k \alpha_C^k) - (\rho_C^k \alpha_C^k)^\circ}{\Delta t} V_C + \sum_{f \sim nb(C)} \dot{m}_f^k \alpha_f^k + \sum_{f \sim nb(C)} \rho_f^k \alpha_f^k (1 - \alpha_f^k) \mathbf{v}_{r_f} \cdot \mathbf{S}_f = 0 \tag{4.34}$$

where $\dot{m}_f^k$ is the mass flux of phase k through the face f, calculated as:

$$\dot{m}_f^k = \rho_f^k \mathbf{v}_f \cdot \mathbf{S}_f \tag{4.35}$$

The algebraic form of the equation is:

$$a_C^{\alpha^k} \alpha_C^k + \sum_{F \sim NB(C)} a_F^{\alpha^k} \alpha_F^k = b_C^{\alpha^k} \tag{4.36}$$

where

$$\begin{aligned} a_C^{\alpha^k} &= \frac{(\rho_C^k)^*}{\Delta t} V_C + \sum_{f \sim nb(C)} ||(\dot{m}_f^k)^*, 0|| \\ a_F^{\alpha^k} &= -|| - (\dot{m}_f^k)^*, 0|| \\ b_C^{\alpha^k} &= \frac{(\rho_C^k \alpha_C^k)^\circ}{\Delta t} V_C - \sum_{f \sim nb(C)} \rho_f^k \alpha_f^k (1 - \alpha_f^k) \mathbf{v}_{r_f} \cdot \mathbf{S}_f \end{aligned}$$

The semi-discrete anti-diffusion term (last term on the left-hand-side) is discussed here. This artificial term is active in the vicinity of the interface region only due to the term $\alpha^k(1-\alpha^k)$ in eq. 4.34. Refs. [17, 18, 19] have mentioned the use of a decoupled compression term from the advection of α^k. The relative velocity $\mathbf{v}_{r_f}$, which acts normal to the interface, is determined such as:

$$\mathbf{v}_{r_f} = \gamma ||\mathbf{v}_f|| \hat{\mathbf{n}} \tag{4.37}$$

where

$$\hat{\mathbf{n}} = \frac{\nabla \alpha_f^k}{||\nabla \alpha_f^k|| + \delta} \tag{4.38}$$

and γ is a compression factor varying between 0 and 1. The value of 0 means that the interface is not compressed, while the value of 1 means that there's maximum compression. Further manipulation for the anti-diffusion term leads to a more implicit formulation. δ is a stabilizing factor, taken to be 10^{-5}, to avoid dividing by 0. Other methods for treating the term are stated in the appendix.

On the other hand, and as mentioned earlier, there are different ways of approaching the solution of the volume fraction equations. This is addressed in what follows.

Explicit Geometric Conservation Constraint

In this approach, $n-1$ volume fraction equations are solved sequentially. The corresponding discrete algebraic equation of an arbitrary volume fraction field k:

$$\forall\ k \in [1, n-1], \qquad a_C^{\alpha^k} \alpha_C^k + \sum_{F \sim NB(C)} a_F^{\alpha^k} \alpha_F^k = b_C^{\alpha^k} \tag{4.39}$$

The n^{th} volume fraction field is computed from the geometric conservation equation

$$\alpha^n = 1 - \sum_{k=1}^{n-1} \alpha^k \tag{4.40}$$

The drawback of this procedure is that the volume fraction field of any phase will not be influenced by the volume fractions fields of other phases except when calculating the n^{th} phase.

Mutual Influence of Volume Fractions

This method is an improved approach. n volume fraction equations for n phases are solved sequentially, so, the algebraic equations are constructed and solved accordingly:

$$\forall\ k \in [1, n], \qquad a_C^{\alpha^k} \alpha_C^k + \sum_{F \sim NB(C)} a_F^{\alpha^k} \alpha_F^k = b_C^{\alpha^k} \tag{4.41}$$

Then, geometric conservation is enforced on the resulting volume fraction values using the following equation

$$\alpha^k := \frac{\alpha^k}{\sum\limits_{l=1}^{n} \alpha^l} \tag{4.42}$$

The latter step is either applied before the assembly and solution of the conservation equations, or after solving the volume fraction equations.

Mutual Influence of Volume Fractions with Implicit Normalization

An implicit normalization procedure is suggested in this work. It is incorporated in a coupled volume fraction discrete system which ensures geometric conservation, with no need to an explicit normalization step. The implicit normalization procedure incorporates the geometric conservation principle into the volume fraction equations by applying appropriate modifications to the coefficients of the discrete algebraic equations. The method is derived starting from the discrete algebraic equation

$$a_C^{\alpha^k} \alpha_C^k + \sum_{F \sim NB(C)} a_F^{\alpha^k} \alpha_F^k = b_C^{\alpha^k} \tag{4.43}$$

Re-arranging

$$\alpha_C^k = \frac{-\sum\limits_{F\sim NB(C)} a_F^{\alpha^k}\alpha_F^k + b_C^{\alpha^k}}{a_C^{\alpha^k}} \tag{4.44}$$

For a two-phase flow, the geometric constraint is written as

$$\alpha_C^1 + \alpha_C^2 = 1 = \frac{-\sum\limits_{F\sim NB(C)} a_F^{\alpha^1}\alpha_F^1 + b_C^{\alpha^1}}{a_C^{\alpha^1}} + \frac{-\sum\limits_{F\sim NB(C)} a_F^{\alpha^2}\alpha_F^2 + b_C^{\alpha^2}}{a_C^{\alpha^2}} \tag{4.45}$$

which yields

$$a_C^{\alpha^1} = \frac{a_C^{\alpha^2}\left(-\sum\limits_{F\sim NB(C)} a_F^{\alpha^1}\alpha_F^1 + b_C^{\alpha^1}\right) + a_C^{\alpha^1}\left(-\sum\limits_{F\sim NB(C)} a_F^{\alpha^2}\alpha_F^2 + b_C^{\alpha^2}\right)}{a_C^{\alpha^2}} \tag{4.46}$$

substituting in the phasic equation 4.44

$$\alpha_C^1 = \frac{-\sum\limits_{F\sim NB(C)} a_F^{\alpha^1}\alpha_F^1 + b_C^{\alpha^1}}{a_C^{\alpha^1}\left(\frac{-\sum\limits_{F\sim NB(C)} a_F^{\alpha^1}\alpha_F^1 + b_C^{\alpha^1}}{a_C^{\alpha^1}} + \frac{-\sum\limits_{F\sim NB(C)} a_F^{\alpha^2}\alpha_F^2 + b_C^{\alpha^2}}{a_C^{\alpha^2}}\right)} \tag{4.47}$$

The residual of the volume fraction equation for a phase k at an element C is defined as

$$R_C^{\alpha^k} = b_C^{\alpha^k} - a_C^{\alpha^k}\alpha_C^k - \sum_{F\sim NB(C)} a_F^{\alpha^k}\alpha_F^k \tag{4.48}$$

incorporating D.68 in 4.47

$$\alpha_C^1 = \frac{-\sum\limits_{F\sim NB(C)} a_F^{\alpha^1}\alpha_F^1 + b_C^{\alpha^1}}{a_C^{\alpha^1}\left(1 - \frac{R_C^{\alpha^1}}{a_C^{\alpha^1}} - \frac{R_C^{\alpha^2}}{a_C^{\alpha^2}}\right)} \tag{4.49}$$

and for a general phase k

$$\alpha_C^k = \frac{-\sum\limits_{F\sim NB(C)} a_F^{\alpha^k}\alpha_F^k + b_C^{\alpha^k}}{a_C^{\alpha^k}\left(1 - \sum\limits_{l=1}^{n} \frac{R_C^{\alpha^l}}{a_C^{\alpha^l}}\right)} \tag{4.50}$$

Now the volume fraction equation of phase k is modified as

$$a_C^{\alpha^k,*}\alpha_C^k + \sum_{F\sim NB(C)} a_F^{\alpha^k}\alpha_F^k = b_C^{\alpha^k} \tag{4.51}$$

where

$$a_C^{\alpha^k,*} = a_C^{\alpha^k} \left(1 - \sum_{l=1}^{n} \frac{R_C^{\alpha^k}}{a_C^{\alpha^k}} \right) \tag{4.52}$$

The discrete algebraic system of the volume fraction equations are n coupled equations, solved simultaneously as

$$\begin{bmatrix} a_C^{\alpha^k,*} & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & a_C^{\alpha^k,*} \end{bmatrix} \begin{bmatrix} \alpha_C^1 \\ \vdots \\ \alpha_C^k \end{bmatrix} + \sum_{F \sim NB(C)} \begin{bmatrix} a_F^{\alpha^k} & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & a_F^{\alpha^k} \end{bmatrix} \begin{bmatrix} \alpha_F^1 \\ \vdots \\ \alpha_F^n \end{bmatrix} = \begin{bmatrix} b_C^{\alpha^1} \\ \vdots \\ b_C^{\alpha^n} \end{bmatrix} \tag{4.53}$$

$$\tag{4.54}$$

4.4 Coupled VOF System

The fully-coupled VOF system discretization is presented in this section. The procedure includes velocity, pressure and volume fraction equations in a single algebraic system as will be shown. The fully-coupled discrete algebraic system over an element C in the computational domain is given by

$$\begin{bmatrix} a_C^{uu} & a_C^{uv} & a_C^{uw} & a_C^{up} & a_C^{u\alpha^1} & \dots & a_C^{u\alpha^k} & \dots & a_C^{u\alpha^n} \\ a_C^{vu} & a_C^{vv} & a_C^{vw} & a_C^{vp} & a_C^{v\alpha^1} & \dots & a_C^{v\alpha^k} & \dots & a_C^{v\alpha^n} \\ a_C^{wu} & a_C^{wv} & a_C^{ww} & a_C^{wp} & a_C^{w\alpha^1} & \dots & a_C^{w\alpha^k} & \dots & a_C^{w\alpha^n} \\ a_C^{pu} & a_C^{pv} & a_C^{pw} & a_C^{pp} & a_C^{p\alpha^1} & \dots & a_C^{p\alpha^k} & \dots & a_C^{p\alpha^n} \\ a_C^{\alpha^1 u} & a_C^{\alpha^1 v} & a_C^{\alpha^1 w} & a_C^{\alpha^1 p} & a_C^{\alpha^1\alpha^1} & \dots & a_C^{\alpha^1\alpha^k} & \dots & a_C^{\alpha^1\alpha^n} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \\ a_C^{\alpha^k u} & a_C^{\alpha^k v} & a_C^{\alpha^k w} & a_C^{\alpha^k p} & a_C^{\alpha^k\alpha^1} & \dots & a_C^{\alpha^k\alpha^k} & \dots & a_C^{\alpha^k\alpha^n} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \\ a_C^{\alpha^n u} & a_C^{\alpha^n v} & a_C^{\alpha^n w} & a_C^{\alpha^n p} & a_C^{\alpha^n\alpha^1} & \dots & a_C^{\alpha^n\alpha^k} & \dots & a_C^{\alpha^n\alpha^n} \end{bmatrix} \begin{bmatrix} u_C \\ v_C \\ w_C \\ p_C \\ \alpha_C^1 \\ \vdots \\ \alpha_C^k \\ \vdots \\ \alpha_C^n \end{bmatrix} \tag{4.55}$$

(4.56)

$$+ \sum_{F\sim NB(C)} \begin{bmatrix} a_F^{uu} & a_F^{uv} & a_F^{uw} & a_F^{up} & a_F^{u\alpha^1} & \dots & a_F^{u\alpha^k} & \dots & a_F^{u\alpha^n} \\ a_F^{vu} & a_F^{vv} & a_F^{vw} & a_F^{vp} & a_F^{v\alpha^1} & \dots & a_F^{v\alpha^k} & \dots & a_F^{v\alpha^n} \\ a_F^{wu} & a_F^{wv} & a_F^{ww} & a_F^{wp} & a_F^{w\alpha^1} & \dots & a_F^{w\alpha^k} & \dots & a_F^{w\alpha^n} \\ a_F^{pu} & a_F^{pv} & a_F^{pw} & a_F^{pp} & a_F^{p\alpha^1} & \dots & a_F^{p\alpha^k} & \dots & a_F^{p\alpha^n} \\ a_F^{\alpha^1 u} & a_F^{\alpha^1 v} & a_F^{\alpha^1 w} & a_F^{\alpha^1 p} & a_F^{\alpha^1\alpha^1} & \dots & a_F^{\alpha^1\alpha^k} & \dots & a_F^{\alpha^1\alpha^n} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \\ a_F^{\alpha^k u} & a_F^{\alpha^k v} & a_F^{\alpha^k w} & a_F^{\alpha^k p} & a_F^{\alpha^k\alpha^1} & \dots & a_F^{\alpha^k\alpha^k} & \dots & a_F^{\alpha^k\alpha^n} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \\ a_F^{\alpha^n u} & a_F^{\alpha^n v} & a_F^{\alpha^n w} & a_F^{\alpha^n p} & a_F^{\alpha^n\alpha^1} & \dots & a_F^{\alpha^n\alpha^k} & \dots & a_F^{\alpha^n\alpha^n} \end{bmatrix} \begin{bmatrix} u_F \\ v_F \\ w_F \\ p_F \\ \alpha_F^1 \\ \vdots \\ \alpha_F^k \\ \vdots \\ \alpha_F^n \end{bmatrix} = \begin{bmatrix} b_C^u \\ b_C^v \\ b_C^w \\ b_C^p \\ b_C^{\alpha^1} \\ \vdots \\ b_C^{\alpha^k} \\ \vdots \\ b_C^{\alpha^n} \end{bmatrix} \tag{4.57}$$

4.4.1 Momentum Conservation Equation

Considering the semi-discrete form of the momentum equation (eq. 4.17), after substituting the body force $\mathbf{B}$ by the gravitational body force $(\rho - \rho_0)\mathbf{g}$, and disregarding any other forms of body forces:

$$\begin{aligned} &\frac{\rho_C \mathbf{v}_C - (\rho_C \mathbf{v}_C)^\circ}{\Delta t} V_C + \sum_{f\sim nb(C)} \dot{m}_f \mathbf{v}_f \\ &= \sum_{f\sim nb(C)} \mu_f \left(\nabla \mathbf{v}_f + \nabla \mathbf{v}_f^T \right) \cdot \mathbf{S}_f - \sum_{f\sim nb(C)} p_f \mathbf{S}_f + (\rho_C - \rho_0)\mathbf{g} V_C \end{aligned}$$

Substituting further the mixture density ρ by its phasic constituents (eq. 3.4) in the gravitational body force term:

$$\begin{aligned}
&\frac{\rho_C \mathbf{v}_C - (\rho_C \mathbf{v}_C)^{\circ}}{\Delta t} V_C + \sum_{f\sim nb(C)} \dot{m}_f \mathbf{v}_f \\
&= \sum_{f\sim nb(C)} \mu_f \left(\nabla \mathbf{v}_f + \nabla \mathbf{v}_f^T\right) \cdot \mathbf{S}_f - \sum_{f\sim nb(C)} p_f \mathbf{S}_f + \left(\sum_{k=1}^{n} \alpha_C^k \rho_C^k - \rho_0\right) \mathbf{g} V_C
\end{aligned}$$

A linear coupling exists in the gravitational body force term and needs no manipulation. The algebraic form of the discrete momentum equations are:

$$\begin{aligned}
&a_C^{uu} u_C + \underline{a_C^{uv} v_C} + \underline{a_C^{uw} w_C} + a_C^{up} p_C + \sum_{k=1}^{n} a_C^{u\alpha^k} \alpha_C^k + \\
&\sum_{F\sim NB(C)} a_F^{uu} u_F + \underline{\sum_{F\sim NB(C)} a_F^{uv} v_F} + \underline{\sum_{F\sim NB(C)} a_F^{uw} w_F} + \sum_{F\sim NB(C)} a_F^{up} p_F + \sum_{k=1}^{n} \sum_{F\sim NB(C)} a_F^{u\alpha^k} \alpha_F^k = b_C^u
\end{aligned}$$

$$\begin{aligned}
&\underline{a_C^{vu} u_C} + a_C^{vv} v_C + \underline{a_C^{vw} w_C} + a_C^{vp} p_C + \sum_{k=1}^{n} a_C^{v\alpha^k} \alpha_C^k + \\
&\underline{\sum_{F\sim NB(C)} a_F^{vu} v_F} + \sum_{F\sim NB(C)} a_F^{vv} v_F + \underline{\sum_{F\sim NB(C)} a_F^{vw} w_F} + \sum_{F\sim NB(C)} a_F^{vp} p_F + \sum_{k=1}^{n} \sum_{F\sim NB(C)} a_F^{v\alpha^k} \alpha_F^k = b_C^v
\end{aligned}$$

$$\begin{aligned}
&\underline{a_C^{wu} u_C} + \underline{a_C^{wv} v_C} + a_C^{ww} w_C + a_C^{wp} p_C + \sum_{k=1}^{n} a_C^{w\alpha^k} \alpha_C^k + \\
&\underline{\sum_{F\sim NB(C)} a_F^{wu} v_F} + \underline{\sum_{F\sim NB(C)} a_F^{wv} v_F} + \sum_{F\sim NB(C)} a_F^{ww} w_F + \sum_{F\sim NB(C)} a_F^{wp} p_F + \sum_{k=1}^{n} \sum_{F\sim NB(C)} a_F^{w\alpha^k} \alpha_F^k = b_C^w
\end{aligned}$$

where

$$\begin{aligned}
a_C^{uu} &= a_C^{vv} = a_C^{ww} = \frac{\rho_C^*}{\Delta t} V_C + \sum_{f\sim nb(C)} \left(||\dot{m}_f^*, 0|| + \mu_f^* \frac{E_f}{d_{CF}} \right) \\
a_F^{uu} &= a_F^{vv} = a_F^{ww} = -|| - \dot{m}_f^*, 0|| - \mu_f^* \frac{E_f}{d_{CF}}
\end{aligned}$$

$$\begin{aligned}
a_C^{up} &= \sum_{f\sim nb(C)} g_f S_f^x & a_C^{vp} &= \sum_{f\sim nb(C)} g_f S_f^y & a_C^{wp} &= \sum_{f\sim nb(C)} g_f S_f^z \\
a_F^{up} &= (1 - g_f) S_f^x & a_F^{vp} &= (1 - g_f) S_f^y & a_F^{wp} &= (1 - g_f) S_f^z
\end{aligned}$$

$$\begin{aligned}
a_C^{u\alpha^k} &= \frac{(\rho_C^k u_C)^*}{\Delta t} V_C - (\rho_C^k)^* g^x V_C \\
a_C^{v\alpha^k} &= \frac{(\rho_C^k v_C)^*}{\Delta t} V_C - (\rho_C^k)^* g^y V_C \\
a_C^{w\alpha^k} &= \frac{(\rho_C^k w_C)^*}{\Delta t} V_C - (\rho_C^k)^* g^z V_C
\end{aligned}$$

$$\begin{aligned}
a_F^{u\alpha^k} &= 0 \\
a_F^{v\alpha^k} &= 0 \\
a_F^{w\alpha^k} &= 0
\end{aligned}$$

$$\begin{aligned}
b_C^u &= \sum_{f \sim nb(C)} \left[\mu_f^* \nabla u_f^* \cdot \mathbf{T}_f\right] + \frac{(\rho_C u_C)^\circ}{\Delta t} V_C + \sum_{f \sim nb(C)} \left[(\mu_f \nabla \mathbf{v}_f^T \cdot \mathbf{S}_f)^* \cdot \hat{\mathbf{i}}\right] - \rho_0 g^x V_C \\
b_C^v &= \sum_{f \sim nb(C)} \left[\mu_f^* \nabla v_f^* \cdot \mathbf{T}_f\right] + \frac{(\rho_C v_C)^\circ}{\Delta t} V_C + \sum_{f \sim nb(C)} \left[(\mu_f \nabla \mathbf{v}_f^T \cdot \mathbf{S}_f)^* \cdot \hat{\mathbf{j}}\right] - \rho_0 g^y V_C \\
b_C^w &= \sum_{f \sim nb(C)} \left[\mu_f^* \nabla w_f^* \cdot \mathbf{T}_f\right] + \frac{(\rho_C w_C)^\circ}{\Delta t} V_C + \sum_{f \sim nb(C)} \left[(\mu_f \nabla \mathbf{v}_f^T \cdot \mathbf{S}_f)^* \cdot \hat{\mathbf{k}}\right] - \rho_0 g^z V_C
\end{aligned}$$

As mentioned earlier, the underlined terms are related to velocity components' interactions, which are zero except at walls and symmetry planes.

4.4.2 Mass Conservation Equation

As for the pressure equation, its coefficients presented in the segregated VOF previously still hold as it involves no coupling, however, there are extra zero terms that should be highlighted for consistency. The algebraic form of the discrete pressure equation within the fully-coupled system is:

$$\begin{aligned}
& a_C^{pu} u_C + a_C^{pv} v_C + a_C^{pw} w_C + a_C^{pp} p_C + \sum_{k=1}^{n} a_C^{p\alpha^k} \alpha_C^k \\
& + \sum_{F \sim NB(C)} a_F^{pu} u_F + \sum_{F \sim NB(C)} a_F^{pv} v_F + \sum_{F \sim NB(C)} a_F^{pw} w_F + \sum_{F \sim NB(C)} a_F^{pp} p_F + \sum_{k=1}^{n} \sum_{F \sim NB(C)} a_F^{p\alpha^k} \alpha_F^k = b_C^p
\end{aligned}$$

where

$$\begin{aligned}
a_C^{p\alpha^k} &= 0 \\
a_F^{p\alpha^k} &= 0
\end{aligned}$$

4.4.3 Volume Fraction Equation

The semi-discrete form of the volume fraction equation (eq. 4.34) is re-written here while highlighting the non-linear coupling term. Substituting eq. 4.35 in eq. 4.34 and disregarding the interfacial anti-diffusion within the current reasoning:

$$\frac{(\rho_C^k \alpha_C^k) - (\rho_C^k \alpha_C^k)^\circ}{\Delta t} V_C + \underbrace{\sum_{f \sim nb(C)} \rho_f^k \mathbf{v}_f \cdot \mathbf{S}_f \alpha_f^k}_{non-linear\ coupling} = 0 \tag{4.58}$$

In the above equation a non-linear coupling between α^k and $\mathbf{v}_f$ exists; a non-linear coupling term is linearized using a Newton-Raphson technique rather than a Picard linearisation technique. A separate section in the appendix is devoted to prove the superiority of the Newton-Raphson linearisation. By all means, a Newton-Raphson linerisation states that a non-linear term $\phi\psi$ is linearized as

$$\phi\psi = \phi^* \psi + \phi \psi^* - \phi^* \psi^* \tag{4.59}$$

Therefore, eq. 4.58 becomes

$$\frac{(\rho_C^k \alpha_C^k) - (\rho_C^k \alpha_C^k)^\circ}{\Delta t} V_C + \sum_{f \sim nb(C)} \left[(\alpha_f^k \rho_f^k)^* \mathbf{v}_f \cdot \mathbf{S}_f + \alpha_f^k (\rho_f^k \mathbf{v}_f)^* \cdot \mathbf{S}_f - (\alpha_f^k \rho_f^k \mathbf{v}_f)^* \cdot \mathbf{S}_f \right] = 0$$

re-arranging:

$$\frac{(\rho_C^k \alpha_C^k) - (\rho_C^k \alpha_C^k)^\circ}{\Delta t} V_C + \sum_{f \sim nb(C)} \left[(\alpha_f^k \rho_f^k)^* \mathbf{v}_f \cdot \mathbf{S}_f + (\rho_f^k \mathbf{v}_f)^* \cdot \mathbf{S}_f \alpha_f^k \right] = \sum_{f \sim nb(C)} (\alpha_f^k \rho_f^k \mathbf{v}_f)^* \cdot \mathbf{S}_f$$

substituting $(\rho_f^k \mathbf{v}_f)^* \cdot \mathbf{S}_f$ by $(\dot{m}_f^k)^*$:

$$\frac{(\rho_C^k \alpha_C^k) - (\rho_C^k \alpha_C^k)^\circ}{\Delta t} V_C + \sum_{f \sim nb(C)} \left[(\alpha_f^k \rho_f^k)^* \mathbf{v}_f \cdot \mathbf{S}_f + (\dot{m}_f^k)^* \alpha_f^k \right] = \sum_{f \sim nb(C)} (\alpha_f^k \dot{m}_f^k)^*$$

α_f^k is advected to the face f, so an upwind or higher resolution advection scheme should be used. However, $\mathbf{v}_f$ is the Rhie-Chow interpolated $\mathbf{v}$:

$$\mathbf{v}_f = \overline{\mathbf{v}_f} - \overline{\mathbf{D}_f^{\mathbf{v}}} \left(\nabla p_f - \overline{\nabla p_f} \right) \tag{4.60}$$

substituting in the equation:

$$\begin{aligned} \frac{(\rho_C^k \alpha_C^k) - (\rho_C^k \alpha_C^k)^\circ}{\Delta t} V_C + \sum_{f \sim nb(C)} \left[(\alpha_f^k \rho_f^k)^* \left(\overline{\mathbf{v}_f} - \overline{\mathbf{D}_f^{\mathbf{v}}} \left(\nabla p_f - \overline{\nabla p_f} \right) \right) \cdot \mathbf{S}_f + (\dot{m}_f^k)^* \alpha_f^k \right] \\ = \sum_{f \sim nb(C)} (\alpha_f^k \dot{m}_f^k)^* \end{aligned}$$

Further re-arrangement:

$$\frac{(\rho_C^k\alpha_C^k)-(\rho_C^k\alpha_C^k)^\circ}{\Delta t}V_C + \underbrace{\sum_{f\sim nb(C)}(\alpha_f^k\rho_f^k)^*\overline{\mathbf{v}_f}\cdot\mathbf{S}_f}_{\alpha^k-\mathbf{v}\ coupling} - \underbrace{\sum_{f\sim nb(C)}(\alpha_f^k\rho_f^k)^*\overline{\mathbf{D}_f^{\mathbf{v}}}\nabla p_f\cdot\mathbf{S}_f}_{\alpha^k-p\ coupling}$$

$$+\sum_{f\sim nb(C)}(\alpha_f^k\rho_f^k)^*\overline{\mathbf{D}_f^{\mathbf{v}}}\;\overline{\nabla p_f}\cdot\mathbf{S}_f + \sum_{f\sim nb(C)}(\dot{m}_f^k)^*\alpha_f^k = \sum_{f\sim nb(C)}(\alpha_f^k\dot{m}_f^k)^*$$

Without any further details, the algebraic form of the discrete volume fraction equations:

$$a_C^{\alpha^k u}u_C + a_C^{\alpha^k v}v_C + a_C^{\alpha^k w}w_C + a_C^{\alpha^k p}p_C + a_C^{\alpha^k\alpha^k}\alpha_C^k$$

$$+\sum_{F\sim NB(C)}a_F^{\alpha^k u}u_F + \sum_{F\sim NB(C)}a_F^{\alpha^k v}v_F + \sum_{F\sim NB(C)}a_F^{\alpha^k w}w_F + \sum_{F\sim NB(C)}a_F^{\alpha^k p}p_F + \sum_{F\sim NB(C)}a_F^{\alpha^k\alpha^k}\alpha_F^k = b_C^{\alpha}$$

where

$$\begin{aligned}
a_C^{\alpha^k u} &= \sum_{f\sim nb(C)}(\alpha_f^k\rho_f^k)^*g_fS_f^x\\
a_C^{\alpha^k v} &= \sum_{f\sim nb(C)}(\alpha_f^k\rho_f^k)^*g_fS_f^y\\
a_C^{\alpha^k w} &= \sum_{f\sim nb(C)}(\alpha_f^k\rho_f^k)^*g_fS_f^z
\end{aligned}$$

$$\begin{aligned}
a_C^{\alpha^k p} &= \sum_{f\sim nb(C)}(\alpha_f^k\rho_f^k)^*\mathcal{D}_f\\
a_C^{\alpha^k\alpha^k} &= \frac{(\rho_C^k)^*}{\Delta t}V_C + \sum_{f\sim nb(C)}||(\dot{m}_f^k)^*,0||
\end{aligned}$$

$$\begin{aligned}
a_F^{\alpha^k u} &= (\alpha_f^k\rho_f^k)^*(1-g_f)S_f^x\\
a_F^{\alpha^k v} &= (\alpha_f^k\rho_f^k)^*(1-g_f)S_f^y\\
a_F^{\alpha^k w} &= (\alpha_f^k\rho_f^k)^*(1-g_f)S_f^z
\end{aligned}$$

$$\begin{aligned}
a_F^{\alpha^k p} &= -(\alpha_f^k\rho_f^k)^*\mathcal{D}_f\\
a_F^{\alpha^k\alpha^k} &= -||-(\dot{m}_f^k)^*,0||
\end{aligned}$$

$$
\begin{aligned}
b_C^{\alpha^k} = \frac{(\rho_C^k \alpha_C^k)^\circ}{\Delta t} V_C + \sum_{f\sim nb(C)} (\alpha_f^k \dot{m}_f^k)^* + \sum_{f\sim nb(C)} (\alpha_f^k \rho_f^k)^* \nabla p_f^* \cdot \mathbf{T}'_f \\
- \sum_{f\sim nb(C)} (\alpha_f^k \rho_f^k)^* \, \overline{\nabla p_f} \cdot \mathbf{S}'_f
\end{aligned}
$$

Worth mentioning at this point is that the reasoning of the implicit normalization of the volume fraction fields in order to maintain geometric conservation may also be incorporated in the fully coupled VOF system by simply substituting the diagonal coefficients of the volume fraction discrete algebraic equation $a_C^{\alpha^k \alpha^k}$ by $a_C^{\alpha^k \alpha^k,*}$ expressed in the formula of eq. 4.52.

4.5 Force Balancing Methods at the Onset of Interface Instability

A very common weakness of free surface flow simulation is the discontinuity of properties at the free surface. The direct consequence of that is the introduction of a discontinuity in the pressure gradient, knowing that the pressure gradient balances body-forces, i.e. gravitational forces [79]. This problem is very prominent in the case of collocated grids because they lead to a weak coupling between velocity and pressure fields [80, 81, 82]. This dilemma has been widely addressed in numerous publications [83, 84]. There are multiple methods for balancing the body force, including the modification of the pressure gradient formula through interpolating the pressure to the face using a density-weighted harmonic interpolation rather than a linear interpolation method, the incorporation of a shifted pressure, and the modification of the Rhie-Chow formula by reconstructing the body-forces in the momentum equation and altering the Rhie-Chow interpolation. These methods are addressed next.

4.5.1 Modified Pressure Gradient

The conventional way to determine the pressure gradient is using the Gauss formula or the least squares method, but it has been shown by [84] that both of them could produce incorrect results in cases where large discontinuity in properties exist. For consistency, the Gauss formula for computing the pressure gradient is:

$$
\nabla p_C = \frac{1}{V_C} \sum_{f\sim nb(C)} p_f \mathbf{S}_f \tag{4.61}
$$

where p_f is commonly interpolated to the face using a linear interpolation. In what follows, modified formulations of the Gauss formula for the computation of the pressure gradient are presented.

Density-Based Harmonic Interpolation of Pressure Field

Zhang et al. [85] reported the use of a harmonic-type interpolation based on the mixture density ρ for pressure field to ensure the mixture acceleration produced by the pressure gradient is continuous across the element faces:

$$p_f = \rho_f \left[g_f \frac{p_C}{\rho_C} + (1 - g_f) \frac{p_F}{\rho_F} \right] \tag{4.62}$$

where ρ_f is harmonically interpolated such that:

$$\rho_f = \frac{1}{\frac{g_f}{\rho_C} + \frac{1-g_f}{\rho_F}} \tag{4.63}$$

Blended Linear-Harmonic Interpolation of Pressure Field

Khrabry et al. [86] have also reported in their work the use of a modified Gauss formula, where they interpolate the pressure to the element face using a blended expression of linear and harmonic interpolation:

$$p_f = \zeta \; p_{f,har} + (1 - \zeta) \; p_{f,lin} \tag{4.64}$$

where,

$$p_{f,lin} = g_f p_C + (1 - g_f) p_F, \qquad p_{f,har} = \rho_f \left[g_f \frac{p_C}{\rho_C} + (1 - g_f) \frac{p_F}{\rho_F} \right] \tag{4.65}$$

$\zeta = \cos^2 \theta$, is the blending factor and θ is the angle between the normal of the face and the direction of gravity:

$$\theta = \arccos \left(\frac{\mathbf{S}_f \cdot \mathbf{g}}{\|\mathbf{S}_f\| \|\mathbf{g}\|} \right) \tag{4.66}$$

Joint Expression of Pressure Gradient and Body-Force

The harmonic and blended linear-harmonic interpolation methods of pressure field are only convenient for structured-type grids, while it was proved that the use of these interpolations for unstructured grids produces unsatisfactory results. To this end, Efremov et al. [10] developed a joint expression for the pressure gradient and the body force term.

4.5.2 Modified Pressure

The motivation of this method emerges from the fact that the pressure field has to be interpolated to the faces to compute the gradient, so a modified (shifted) pressure can be obtained by excluding the hydrostatic force from the pressure. This is not to be confused with the pressure shift due to reference density. However, this shift of pressure removes the discontinuous step gravity term, and produces a pressure field that is continuous instead of piecewise continuous for quiescent flows, and this in turn allows for simple interpolation to the element face [20, 87]. Modifying the pressure field by excluding hydrostatic term has also the advantage of simplifying the definition of pressure at the boundaries in the VOF method [88]. Denoting the shifted pressure by p', the absolute pressure is decomposed as:

$$p = p' + (\rho - \rho_0)\mathbf{g} \cdot (\mathbf{R} - \mathbf{R}_{ref}) \tag{4.67}$$

Applying gradient operator on both sides:

$$\nabla p = \nabla p' + (\rho - \rho_0)\mathbf{g} + \mathbf{g} \cdot (\mathbf{R} - \mathbf{R}_{ref})\nabla \rho \tag{4.68}$$

Substituting ∇p in the Navier-Stokes equation 3.1 with the above expression, and replacing $\mathbf{B}$ in the Navier-Stokes equation with $(\rho - \rho_0)\mathbf{g}$ lead to the following form of the equation:

$$\frac{\partial \rho \mathbf{v}}{\partial t} + \nabla \cdot (\rho \mathbf{v}\mathbf{v}) = \nabla \cdot \tau - \nabla p' - \mathbf{g} \cdot (\mathbf{R} - \mathbf{R}_{ref})\nabla \rho \tag{4.69}$$

The Rhie-Chow interpolation is modified so as to account for the modification in the momentum equation as presented by [72]:

$$\mathbf{v}_f = \overline{\mathbf{v}_f} - \overline{\mathbf{D}^{\mathbf{v}}_f}\left(\nabla p'_f - \overline{\nabla p'_f}\right) - \overline{\mathbf{D}^{\mathbf{v}}_f}\left[\mathbf{g} \cdot (\mathbf{R}_f - \mathbf{R}_{ref})\right]\left(\nabla \rho_f - \overline{\nabla \rho_f}\right) \tag{4.70}$$

4.5.3 Modifications to Rhie-Chow Interpolation

The simulation of free surface flows requires further modification to the Rhie-Chow interpolation in order to properly calculate the face velocity, which is advecting the phases [89, 90, 91]. The modification of the Rhie-Chow interpolation involves the proper treatment of body-forces and a better element-to-face interpolation formula for the explicit terms in the Rhie-Chow formula.

Modifications Related to Body-Force

The Rhie-Chow interpolation, which was critical to eliminate oscillations in velocity and pressure fields in solving flows over collocated grids, has not considered strong discontinuities in the body force field [92]. In this section the method of reconstructing the body force to make it inline with the Rhie-Chow formula is stated. Denoting all the existing body forces by $\mathbf{B}_C$, a redistributed body force $\overline{\overline{\mathbf{B}_C}}$ is calculated in two major steps. First, an interpolated body force term at the cell face $\overline{\mathbf{B}_f}$ is calculated by a linear interpolation:

$$\overline{\mathbf{B}_f} = g_f \mathbf{B}_C + (1 - g_f)\mathbf{B}_F \tag{4.71}$$

or by a density-weighted harmonic interpolation as reported by [91, 93]:

$$\overline{\mathbf{B}_f} = \rho_f \left[g_f \frac{\mathbf{B}_C}{\rho_C} + (1 - g_f) \frac{\mathbf{B}_F}{\rho_F} \right] \tag{4.72}$$

where ρ_f is harmonically interpolated such that:

$$\rho_f = \frac{1}{\frac{g_f}{\rho_C} + \frac{1-g_f}{\rho_F}} \tag{4.73}$$

The next step in the derivation of $\overline{\overline{\mathbf{B}_C}}$ is dictated by the fact that the pressure gradient should be in equilibrium with the body forces at rest,

$$\mathbf{B}_C = \nabla p = \frac{\sum\limits_{f \sim nb(C)} p_f \mathbf{S}_f}{V_C} \tag{4.74}$$

p_f is linearly interpolated from straddling elements' values:

$$p_f = g_f p_C + (1 - g_f) p_F$$

At equilibrium, p_C and p_F can be related as follows:

$$p_F = p_C + \mathbf{B}_f \cdot \mathbf{d}_{CF} \tag{4.75}$$

Substituting back in p_f:

$$p_f = g_f p_C + (1 - g_f) \big[p_C + \mathbf{B}_f \cdot \mathbf{d}_{CF} \big]$$

Re-arranging:

$$p_f = p_C + (1 - g_f) \mathbf{B}_f \cdot \mathbf{d}_{CF}$$

Substituting in equation 4.74:

$$\mathbf{B}_C = \frac{\sum\limits_{f \sim nb(C)} \big[p_C + (1 - g_f) \mathbf{B}_f \cdot \mathbf{d}_{CF} \big] \mathbf{S}_f}{V_C} \tag{4.76}$$

Then,

$$\mathbf{B}_C = \frac{\sum\limits_{f \sim nb(C)} p_C \mathbf{S}_f}{V_C} + \frac{\sum\limits_{f \sim nb(C)} (1 - g_f) \mathbf{B}_f \cdot \mathbf{d}_{CF} \mathbf{S}_f}{V_C} \tag{4.77}$$

p_C can be taken out of the summation, and knowing that $\sum_{f \sim nb(C)} \mathbf{S}_f = 0$:

$$\mathbf{B}_C = p_C \underbrace{\frac{\sum\limits_{f \sim nb(C)} \mathbf{S}_f}{V_C}}_{=0} + \frac{\sum\limits_{f \sim nb(C)} (1 - g_f) \mathbf{B}_f \cdot \mathbf{d}_{CF} \mathbf{S}_f}{V_C} \tag{4.78}$$

Thus the redistributed body force $\overline{\overline{\mathbf{B}_C}}$ can be expressed as

$$\overline{\overline{\mathbf{B}_C}} = \frac{\sum\limits_{f\sim nb(C)} (1-g_f)\left(\overline{\mathbf{B}_f}\cdot\mathbf{d}_{CF}\right)\mathbf{S}_f}{V_C} \tag{4.79}$$

Now the extended Rhie-Chow interpolated cell face velocity $\mathbf{v}_f$ is obtained as

$$\mathbf{v}_f = \overline{\mathbf{v}_f} - \overline{\mathbf{D}_f^{\mathbf{v}}}\left(\nabla p_f - \overline{\nabla p_f}\right) + \overline{\mathbf{D}_f^{\mathbf{v}}}\left(\overline{\mathbf{B}_f} - \overline{\overline{\overline{\mathbf{B}_f}}}\right) \tag{4.80}$$

where $\overline{\overline{\overline{\mathbf{B}_f}}}$ is interpolated also using either linear or a density-based harmonic interpolation:

$$\overline{\overline{\overline{\mathbf{B}_f}}} = \rho_f\left[g_f\frac{\overline{\overline{\mathbf{B}_C}}}{\rho_C} + (1-g_f)\frac{\overline{\overline{\mathbf{B}_F}}}{\rho_F}\right] \tag{4.81}$$

It is also possible to determine a balancing formula for the body force at the face $\mathbf{B}_f$, to be used instead of using an interpolated body force $\overline{\mathbf{B}_f}$, which ensures a better balancing as compared to simple interpolation of the element body forces to faces. This is done using similar analysis as that addressed for the redistributed body force method. At static conditions, and using the corrected face-gradient definition:

$$\begin{aligned}
\mathbf{B}_f = \nabla p_f = \overline{\nabla p_f} - \left(\overline{\nabla p_f}\cdot\mathbf{e}_f - \frac{p_F - p_C}{d_{CF}}\right)\mathbf{e}_f \\
= g_f\nabla p_C + (1-g_f)\nabla p_F - \left[(g_f\nabla p_C + (1-g_f)\nabla p_F)\cdot\mathbf{e}_f - \frac{p_F - p_C}{d_{CF}}\right]\mathbf{e}_f
\end{aligned}$$

We have:

$$\begin{aligned}
\nabla p_C &= \overline{\overline{\mathbf{B}_C}} \\
\nabla p_F &= \overline{\overline{\mathbf{B}_F}}
\end{aligned}$$

and $p_C = p_F - (\rho_f - \rho_0)\mathbf{g}\cdot\mathbf{d}_{CF}$. Then, $p_F - p_C = (\rho_f - \rho_0)\mathbf{g}\cdot\mathbf{d}_{CF}$. Substituting back:

$$\begin{aligned}
\mathbf{B}_f &= g_f\overline{\overline{\mathbf{B}_C}} + (1-g_f)\overline{\overline{\mathbf{B}_F}} - \left[(g_f\overline{\overline{\mathbf{B}_C}} + (1-g_f)\overline{\overline{\mathbf{B}_F}})\cdot\mathbf{e}_f - \frac{(\rho_f-\rho_0)\mathbf{g}\cdot\mathbf{d}_{CF}}{d_{CF}}\right]\mathbf{e}_f \\
&= g_f\overline{\overline{\mathbf{B}_C}} + (1-g_f)\overline{\overline{\mathbf{B}_F}} - \left[(g_f\overline{\overline{\mathbf{B}_C}} + (1-g_f)\overline{\overline{\mathbf{B}_F}})\cdot\mathbf{e}_f - (\rho_f-\rho_0)\mathbf{g}\cdot\mathbf{e}_f\right]\mathbf{e}_f \\
&= g_f\overline{\overline{\mathbf{B}_C}} + (1-g_f)\overline{\overline{\mathbf{B}_F}} - \left[g_f\overline{\overline{\mathbf{B}_C}} + (1-g_f)\overline{\overline{\mathbf{B}_F}} - (\rho_f-\rho_0)\mathbf{g}\right]\cdot\mathbf{e}_f\mathbf{e}_f \\
&= \overline{\overline{\overline{\mathbf{B}_f}}} - \left[\overline{\overline{\overline{\mathbf{B}_f}}} - (\rho_f-\rho_0)\mathbf{g}\right]\cdot\mathbf{e}_f\mathbf{e}_f
\end{aligned}$$

The above treatment of the body-force is supposed to increase the robustness of the solution for free-surface flows due to the fact, as mentioned previously, that there exist sharp variations among the phases' densities.

Large-Stencil Face Pressure Gradient in the Rhie-Chow Formula

$\overline{\nabla p_f}$, the large stencil pressure gradient at face, is commonly calculated using a linear interpolation, however, it is stated by [91, 93] that a harmonic interpolation of the large-stencil pressure gradient term in the Rhie-Chow has a stabilizing advantage when density differences among phases are large. The interpolation is expressed as:

$$\overline{\nabla p_f} = \rho_f \left[g_f \frac{\nabla p_C}{\rho_C} + (1 - g_f) \frac{\nabla p_F}{\rho_F} \right] \tag{4.82}$$

where ρ_f is harmonically interpolated such that:

$$\rho_f = \frac{1}{\frac{g_f}{\rho_C} + \frac{1-g_f}{\rho_F}} \tag{4.83}$$

The calculated values for each of the terms are calculated next for the first iteration. Starting with the small-stencil pressure gradient ∇p_f, it is calculated as:

$$\nabla p_f = \overline{\nabla p_f} - \left(\overline{\nabla p_f} \cdot \mathbf{e}_f - \frac{p_F - p_C}{d_{CF}} \right) \mathbf{e}_f \tag{4.84}$$

The large-stencil pressure gradient $\overline{\nabla p_f}$ is determined using a linear interpolation for the straddling elements' pressure gradients:

$$\overline{\nabla p_f} = g_f \nabla p_C + (1 - g_f) \nabla p_F \tag{4.85}$$

and the pressure gradient at an arbitrary element C is calculated using Gauss formula:

$$\nabla p_C = \frac{1}{V_C} \sum_{f \sim nb(C)} p_f \mathbf{S}_f \tag{4.86}$$

The interpolated face pressure p_f is linearly interpolated:

$$p_f = g_f p_C + (1 - g_f) p_F \tag{4.87}$$

4.5.4 Influence of the Body-Force Reconstruction on the Algebraic Equations

The reconstruction of body forces in the momentum according to the last method presented above has a direct influence on the discretization method, while the other methods do not have a direct influence. As a result of reconstructing the body forces, the Navier-Stokes equation

$$\begin{aligned} &\frac{(\rho_C \mathbf{v}_C) - (\rho_C \mathbf{v}_C)^\circ}{\Delta t} V_C + \sum_{f \sim nb(C)} \dot{m}_f \mathbf{v}_f \\ &= \sum_{f \sim nb(C)} \mu_f \left(\nabla \mathbf{v}_f + \nabla \mathbf{v}_f^T \right) \cdot \mathbf{S}_f - \sum_{f \sim nb(C)} p_f \mathbf{S}_f + \overline{\overline{\mathbf{B}_C}} V_C \end{aligned} \tag{4.88}$$

The appearance of a redistributed gravitational body force in the momentum equation will influence the coupling of the equation with the volume fraction equations. Considering the redistribution of the gravitational force:

$$\overline{\overline{(\rho_C-\rho_0)\mathbf{g}}}=\frac{\sum\limits_{f}(1-g_f)\big[(\rho_f-\rho_0)\mathbf{g}\cdot\mathbf{d}_{CF}\big]\mathbf{S}_f}{V_C}\tag{4.89}$$

re-arranging the formula:

$$\overline{\overline{(\rho_C-\rho_0)\mathbf{g}}}=\sum_{f}(\rho_f-\rho_0)\big[(1-g_f)\left(\mathbf{g}\cdot\mathbf{d}_{CF}\right)\mathbf{S}_f/V_C\big]\tag{4.90}$$

and defining a new gravity-geometry vector quantity at a face $\mathbf{C}_f^{\mathbf{g}}$ such that:

$$\mathbf{C}_f^{\mathbf{g}}=(1-g_f)\left(\mathbf{g}\cdot\mathbf{d}_{CF}\right)\mathbf{S}_f/V_C\tag{4.91}$$

then the redistributed gravitational force is:

$$\overline{\overline{(\rho_C-\rho_0)\mathbf{g}}}=\sum_{f}(\rho_f-\rho_0)\mathbf{C}_f^{\mathbf{g}}\tag{4.92}$$

Now the density at the face may be interpolated using any of the element-face interpolation variants, and each of them will have a specific coupling influence. Only the linear and harmonic interpolations will be presented here. For linear interpolation:

$$\rho_f=g_f\rho_C+(1-g_f)\rho_F$$

substituting by phasic constituents of ρ_f:

$$\rho_f=g_f\left(\sum_{k}\alpha_C^k\rho_C^k\right)+(1-g_f)\left(\sum_{k}\alpha_F^k\rho_F^k\right)$$

then,

$$\overline{\overline{(\rho_C-\rho_0)\mathbf{g}}}=\sum_{f}g_f\left(\sum_{k}\alpha_C^k\rho_C^k\right)\mathbf{C}_f^{\mathbf{g}}+\sum_{f}(1-g_f)\left(\sum_{k}\alpha_F^k\rho_F^k\right)\mathbf{C}_f^{\mathbf{g}}-\sum_{f}\rho_0\mathbf{C}_f^{\mathbf{g}}\tag{4.93}$$

re-arranging:

$$\overline{\overline{(\rho_C-\rho_0)\mathbf{g}}}=\sum_{k}\left[\left(\rho_C^k\sum_{f}g_f\mathbf{C}_f^{\mathbf{g}}\right)\alpha_C^k\right]+\sum_{k}\left\{\sum_{f}\left[\rho_F^k(1-g_f)\mathbf{C}_f^{\mathbf{g}}\alpha_F^k\right]\right\}-\sum_{f}\rho_0\mathbf{C}_f^{\mathbf{g}}\tag{4.94}$$

The coefficients of the momentum equation are modified as follows:

$$\begin{aligned}
a_C^{u\alpha^k} &= \frac{(\rho_C^k u_C)^*}{\Delta t} V_C - (\rho_C^k)^* \sum_{f\sim nb(C)} g_f C_f^{\mathbf{g},x} V_C \\
a_C^{v\alpha^k} &= \frac{(\rho_C^k v_C)^*}{\Delta t} V_C - (\rho_C^k)^* \sum_{f\sim nb(C)} g_f C_f^{\mathbf{g},y} V_C \\
a_C^{w\alpha^k} &= \frac{(\rho_C^k w_C)^*}{\Delta t} V_C - (\rho_C^k)^* \sum_{f\sim nb(C)} g_f C_f^{\mathbf{g},z} V_C
\end{aligned}$$

$$\begin{aligned}
a_F^{u\alpha^k} &= -(\rho_F^k)^*(1-g_f) C_f^{\mathbf{g},x} V_C \\
a_F^{v\alpha^k} &= -(\rho_F^k)^*(1-g_f) C_f^{\mathbf{g},y} V_C \\
a_F^{w\alpha^k} &= -(\rho_F^k)^*(1-g_f) C_f^{\mathbf{g},z} V_C
\end{aligned}$$

On another hand, for a harmonic interpolation:

$$\rho_f = \frac{1}{\frac{g_f}{\rho_C} + \frac{(1-g_f)}{\rho_F}} = \frac{\rho_C \rho_F}{g_f \rho_F + (1-g_f)\rho_C}$$

This interpolation is non-linear, hence, it requires a linearization strategy to express its phasic constituents in a linear version, in order to be able to couple them with the volume fraction equations. Using the general two-variable function linearization presented previously, ρ_f can be expressed as:

$$\begin{aligned}
\rho_f &= \frac{\rho_C \rho_F}{g_f \rho_F + (1-g_f)\rho_C} \\
&= \frac{\rho_C^* \rho_F^*}{g_f \rho_F^* + (1-g_f)\rho_C^*} + \left\{ \frac{\rho_F^* \left[g_f \rho_F^* + (1-g_f)\rho_C^* \right] - (1-g_f)\rho_C^* \rho_F^*}{\left[g_f \rho_F^* + (1-g_f)\rho_C^* \right]^2} \right\} (\rho_C - \rho_C^*) \\
&+ \left\{ \frac{\rho_C^* \left[g_f \rho_F^* + (1-g_f)\rho_C^* \right] - g_f \rho_C^* \rho_F^*}{\left[g_f \rho_F^* + (1-g_f)\rho_C^* \right]^2} \right\} (\rho_F - \rho_F^*)
\end{aligned} \tag{4.95}$$

Simplifying:

$$\rho_f = \left\{ \frac{g_f \rho_F^* \rho_F^*}{\left[g_f \rho_F^* + (1-g_f)\rho_C^* \right]^2} \right\} \rho_C + \left\{ \frac{\rho_C^* (1-g_f) \rho_C^*}{\left[g_f \rho_F^* + (1-g_f)\rho_C^* \right]^2} \right\} \rho_F \tag{4.96}$$

And putting in compact form:

$$\rho_f = C_f^C \rho_C + C_f^F \rho_F \tag{4.97}$$

where

$$\begin{aligned} C_f^C &= \frac{g_f(\rho_F^*)^2}{\left[g_f\rho_F^* + (1-g_f)\rho_C^*\right]^2} \\ C_f^F &= \frac{(1-g_f)(\rho_C^*)^2}{\left[g_f\rho_F^* + (1-g_f)\rho_C^*\right]^2} \end{aligned} \tag{4.98}$$

Substituting by phasic constituents of ρ_C and ρ_F:

$$\rho_f = C_f^C\left(\sum_k \alpha_C^k\rho_C^k\right) + C_f^F\left(\sum_k \alpha_F^k\rho_F^k\right)$$

then,

$$\overline{\overline{(\rho_C-\rho_0)\mathbf{g}}} = \sum_f C_f^C\left(\sum_k \alpha_C^k\rho_C^k\right)\mathbf{C}_f^{\mathbf{g}} + \sum_f C_f^F\left(\sum_k \alpha_F^k\rho_F^k\right)\mathbf{C}_f^{\mathbf{g}} - \sum_f \rho_0\mathbf{C}_f^{\mathbf{g}} \tag{4.99}$$

re-arranging:

$$\overline{\overline{(\rho_C-\rho_0)\mathbf{g}}} = \sum_k\left[\left(\rho_C^k\sum_f C_f^C\mathbf{C}_f^{\mathbf{g}}\right)\alpha_C^k\right] + \sum_k\left\{\sum_f\left[\rho_F^k C_f^F\mathbf{C}_f^{\mathbf{g}}\alpha_F^k\right]\right\} - \sum_f \rho_0\mathbf{C}_f^{\mathbf{g}} \tag{4.100}$$

The coefficients of the momentum equation are modified as follows:

$$\begin{aligned} a_C^{u\alpha^k} &= \frac{(\rho_C^k u_C)^*}{\Delta t}V_C - (\rho_C^k)^*\sum_{f\sim nb(C)} C_f^C C_f^{\mathbf{g},x}V_C \\ a_C^{v\alpha^k} &= \frac{(\rho_C^k v_C)^*}{\Delta t}V_C - (\rho_C^k)^*\sum_{f\sim nb(C)} C_f^C C_f^{\mathbf{g},y}V_C \\ a_C^{w\alpha^k} &= \frac{(\rho_C^k w_C)^*}{\Delta t}V_C - (\rho_C^k)^*\sum_{f\sim nb(C)} C_f^C C_f^{\mathbf{g},z}V_C \end{aligned}$$

$$\begin{aligned} a_F^{u\alpha^k} &= -(\rho_F^k)^* C_f^F C_f^{\mathbf{g},x}V_C \\ a_F^{v\alpha^k} &= -(\rho_F^k)^* C_f^F C_f^{\mathbf{g},y}V_C \\ a_F^{w\alpha^k} &= -(\rho_F^k)^* C_f^F C_f^{\mathbf{g},z}V_C \end{aligned}$$

4.6 The Overdetermined System Dilemma

The fully coupled system presented in the previous section could suffer from a degradation originating from the fact that it might be overdetermined in some cases. In the same system, in addition to the momentum conservation equation, a mass conservation equation and two phasic equations are solved and the volume fractions are then normalized to ensure that their sum is unity. Recalling that the phasic equations make up the mass conservation itself when they are summed together, the system is then said to be overdetermined because the mass conservation is in a way a linear combination of the two phasic equations. To remedy this dilemma which can cause instabilities, or failure at some point, a reformation of the system has to be made. A more ambitious system has to eliminate the mass conservation equation since mass is already conserved on a phasic basis through the two phasic equations. Also, instead of solving the system and normalizing the volume fractions afterwards, it is also applicable to include the geometric conservation equation (3.7), which ensures the unity sum of the volume fractions, in the coupled system. This means that the new system looks like:

$$
\begin{bmatrix}
a_C^{uu} & a_C^{uv} & a_C^{uw} & a_C^{up} & a_C^{u\alpha^1} & \dots & a_C^{u\alpha^k} & \dots & a_C^{u\alpha^n} \\
a_C^{vu} & a_C^{vv} & a_C^{vw} & a_C^{vp} & a_C^{v\alpha^1} & \dots & a_C^{v\alpha^k} & \dots & a_C^{v\alpha^n} \\
a_C^{wu} & a_C^{wv} & a_C^{ww} & a_C^{wp} & a_C^{w\alpha^1} & \dots & a_C^{w\alpha^k} & \dots & a_C^{w\alpha^n} \\
0 & 0 & 0 & 0 & 1 & \dots & 1 & \dots & 1 \\
a_C^{\alpha^1 u} & a_C^{\alpha^1 v} & a_C^{\alpha^1 w} & a_C^{\alpha^1 p} & a_C^{\alpha^1\alpha^1} & \dots & a_C^{\alpha^1\alpha^k} & \dots & a_C^{\alpha^1\alpha^n} \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \\
a_C^{\alpha^k u} & a_C^{\alpha^k v} & a_C^{\alpha^k w} & a_C^{\alpha^k p} & a_C^{\alpha^k\alpha^1} & \dots & a_C^{\alpha^k\alpha^k} & \dots & a_C^{\alpha^k\alpha^n} \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \\
a_C^{\alpha^n u} & a_C^{\alpha^n v} & a_C^{\alpha^n w} & a_C^{\alpha^n p} & a_C^{\alpha^n\alpha^1} & \dots & a_C^{\alpha^n\alpha^k} & \dots & a_C^{\alpha^n\alpha^n}
\end{bmatrix}
\begin{bmatrix}
u_C \\ v_C \\ w_C \\ p_C \\ \alpha_C^1 \\ \vdots \\ \alpha_C^k \\ \vdots \\ \alpha_C^n
\end{bmatrix}
$$

$$
+ \sum_{F\sim NB(C)}
\begin{bmatrix}
a_F^{uu} & a_F^{uv} & a_F^{uw} & a_F^{up} & a_F^{u\alpha^1} & \dots & a_F^{u\alpha^k} & \dots & a_F^{u\alpha^n} \\
a_F^{vu} & a_F^{vv} & a_F^{vw} & a_F^{vp} & a_F^{v\alpha^1} & \dots & a_F^{v\alpha^k} & \dots & a_F^{v\alpha^n} \\
a_F^{wu} & a_F^{wv} & a_F^{ww} & a_F^{wp} & a_F^{w\alpha^1} & \dots & a_F^{w\alpha^k} & \dots & a_F^{w\alpha^n} \\
0 & 0 & 0 & 0 & 0 & \dots & 0 & \dots & 0 \\
a_F^{\alpha^1 u} & a_F^{\alpha^1 v} & a_F^{\alpha^1 w} & a_F^{\alpha^1 p} & a_F^{\alpha^1\alpha^1} & \dots & a_F^{\alpha^1\alpha^k} & \dots & a_F^{\alpha^1\alpha^n} \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \\
a_F^{\alpha^k u} & a_F^{\alpha^k v} & a_F^{\alpha^k w} & a_F^{\alpha^k p} & a_F^{\alpha^k\alpha^1} & \dots & a_F^{\alpha^k\alpha^k} & \dots & a_F^{\alpha^k\alpha^n} \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \\
a_F^{\alpha^n u} & a_F^{\alpha^n v} & a_F^{\alpha^n w} & a_F^{\alpha^n p} & a_F^{\alpha^n\alpha^1} & \dots & a_F^{\alpha^n\alpha^k} & \dots & a_F^{\alpha^n\alpha^n}
\end{bmatrix}
\begin{bmatrix}
u_F \\ v_F \\ w_F \\ p_F \\ \alpha_F^1 \\ \vdots \\ \alpha_F^k \\ \vdots \\ \alpha_F^n
\end{bmatrix}
=
\begin{bmatrix}
b_C^u \\ b_C^v \\ b_C^w \\ 1 \\ b_C^{\alpha^1} \\ \vdots \\ b_C^{\alpha^k} \\ \vdots \\ b_C^{\alpha^n}
\end{bmatrix}
$$

The aforementioned system appears to be consistent. However, there are zeros introduced at the diagonal of the matrix, at the row corresponding to the geometric conservation. This has a direct consequence on the use of the linear solver in the sense that a special iterative linear solver has to be provided such that it considers zero-diagonals; this is a matter of investigation which is not considered here. On another hand, it should be noted here that since the geometric conservation is introduced into the system, the normalization of the volume fractions after solving the system may be skipped because the unity sum of the volume fractions is ensured from the solution of the system. By all means, it has to be clarified that the incorporation of the above system is dependent on the success or failure of the original system.

Chapter 5

Compressibility Considerations

The assumption of incompressible fluids for most free surface flows is very reasonable, because the density changes due to pressure variations are usually negligible [94, 95]. However, some cases involve violent free surface flows such as liquid impact phenomena (liquid sloshing in tank, water impact on flat surface of a wide ship, etc.). These cases often involve air trapping and air cushion effect which are systems with high density ratios that entrap gas pockets; they are subjected to notable fluctuations in pressure. In this case, it is desired to account for compressibility effects which may play an important role in the pressure variations. Unfortunately, the compressibility effect in this context is still not fully addressed in literature [96, 97]. Other free surface flow cases, such as those involving liquid film cooling with a weakly compressible gas, are also good examples which should be regarded in this section. In this work, a compressible VOF model is developed for cases which exhibit low compressibility, such as liquid sloshing in tank. The gas is considered to be an ideal gas, while the liquid is incompressible. So, the proposed solver assumes a single fluid (gas) to be compressible while the rest incompressible.

The compressible fluid flow behavior is determined by the set of equations presented respectively below, namely, Navier-Stokes, continuity, and energy in its total enthalpy form:

$$\frac{\partial \rho \mathbf{v}}{\partial t} + \nabla \cdot (\rho \mathbf{v}\mathbf{v}) = \nabla \cdot \tau - \nabla p + \rho \mathbf{g} \tag{5.1}$$

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \tag{5.2}$$

$$\frac{\partial \rho H}{\partial t} + \nabla \cdot (\rho \mathbf{v} H) = \nabla \cdot (\lambda \nabla T) + \frac{\partial p}{\partial t} + Q^H \tag{5.3}$$

τ here is the stress tensor defined as:

$$\tau = \mu \left(\nabla \mathbf{v} + \nabla \mathbf{v}^T \right) - \frac{2}{3} \mu (\nabla \cdot \mathbf{v}) \mathbf{I} \tag{5.4}$$

λ is the mixture thermal conductivity, defined as:

$$\lambda = \sum_{k=1}^{n} \alpha^k \lambda^k \tag{5.5}$$

Q^H in the total enthalpy equation is any energy source such as viscous dissipation. An additional equation is required to relate density to pressure and temperature. Since one of the phases is a gas phase, and assuming the ideal gas law, the mixture density is formulated as follows:

$$\rho = \sum_{k=1}^{n} \alpha^k \rho^k \tag{5.6}$$

$$= \overbrace{\alpha^1 \rho^1}^{gas\ phase} + \overbrace{\sum_{k=2}^{n} \alpha^k \rho^k}^{liquid\ phases} \tag{5.7}$$

$$= \overbrace{\alpha^1 \frac{p}{R^1 T}}^{gas\ phase} + \overbrace{\sum_{k=2}^{n} \alpha^k \rho^k}^{liquid\ phases} \tag{5.8}$$

The continuity equation 5.2 is re-written here in phasic build-up form for clarity:

$$\sum_{k=1}^{n} \left[\frac{\partial \rho^k \alpha^k}{\partial t} + \nabla \cdot (\rho^k \mathbf{v} \alpha^k) = 0 \right] \tag{5.9}$$

Dividing by a density scale to avoid round-off errors:

$$\sum_{k=1}^{n} \left[\frac{1}{\rho^k_{scale}} \frac{\partial \rho^k \alpha^k}{\partial t} + \frac{1}{\rho^k_{scale}} \nabla \cdot (\rho^k \mathbf{v} \alpha^k) = 0 \right] \tag{5.10}$$

The semi-discrete form of the above equation is expressed at an arbitrary volume C as follows:

$$\sum_{k=1}^{n} \left[\frac{1}{\rho^k_{scale}} \frac{(\rho^k_C \alpha^k_C)^* - (\rho^k_C \alpha^k_C)^\circ}{\Delta t} V_C + \frac{1}{\rho^k_{scale}} \sum_{f \sim nb(C)} \alpha^k_f \rho^k_f \mathbf{v}_f \cdot \mathbf{S}_f = 0 \right] \tag{5.11}$$

Separating the phases into a gas phase and liquid phases:

$$\begin{gathered} \overbrace{\frac{1}{\rho^1_{scale}} \frac{(\rho^1_C \alpha^1_C)^* - (\rho^1_C \alpha^1_C)^\circ}{\Delta t} V_C + \frac{1}{\rho^1_{scale}} \sum_{f \sim nb(C)} \alpha^1_f \rho^1_f \mathbf{v}_f \cdot \mathbf{S}_f}^{gas\ phase} \\ \overbrace{+ \sum_{k=2}^{n} \left[\frac{1}{\rho^k_{scale}} \frac{(\rho^k_C \alpha^k_C)^* - (\rho^k_C \alpha^k_C)^\circ}{\Delta t} V_C + \frac{1}{\rho^k_{scale}} \sum_{f \sim nb(C)} \alpha^k_f \rho^k_f \mathbf{v}_f \cdot \mathbf{S}_f \right]}^{liquid\ phases} = 0 \end{gathered} \tag{5.12}$$

For the gas phase, the density is not constant, so the second term in equation 5.12 can be linearized according to Newton as:

$$
\begin{aligned}
\frac{1}{\rho^1_{scale}} \sum_{f\sim nb(C)} \alpha^1_f \rho^1_f \mathbf{v}_f \cdot \mathbf{S}_f = {} & \frac{1}{\rho^1_{scale}} \sum_{f\sim nb(C)} \alpha^1_f (\rho^1_f)^* \mathbf{v}_f \cdot \mathbf{S}_f \\
& + \frac{1}{\rho^1_{scale}} \sum_{f\sim nb(C)} \alpha^1_f \rho^1_f \mathbf{v}^*_f \cdot \mathbf{S}_f \\
& - \frac{1}{\rho^1_{scale}} \sum_{f\sim nb(C)} \alpha^1_f (\rho^1_f)^* \mathbf{v}^*_f \cdot \mathbf{S}_f
\end{aligned}
\tag{5.13}
$$

Substituting back into the equation:

$$
\begin{aligned}
& \frac{1}{\rho^1_{scale}} \frac{(\rho^1_C \alpha^1_C)^* - (\rho^1_C \alpha^1_C)^\circ}{\Delta t} V_C + \frac{1}{\rho^1_{scale}} \sum_{f\sim nb(C)} \alpha^1_f (\rho^1_f)^* \mathbf{v}_f \cdot \mathbf{S}_f \\
& + \frac{1}{\rho^1_{scale}} \sum_{f\sim nb(C)} \alpha^1_f \rho^1_f \mathbf{v}^*_f \cdot \mathbf{S}_f - \frac{1}{\rho^1_{scale}} \sum_{f\sim nb(C)} \alpha^1_f (\rho^1_f)^* \mathbf{v}^*_f \cdot \mathbf{S}_f \\
& + \sum_{k=2}^{n} \left[\frac{1}{\rho^k_{scale}} \frac{(\rho^k_C \alpha^k_C)^* - (\rho^k_C \alpha^k_C)^\circ}{\Delta t} V_C + \frac{1}{\rho^k_{scale}} \sum_{f\sim nb(C)} \alpha^k_f \rho^k_f \mathbf{v}_f \cdot \mathbf{S}_f \right] = 0
\end{aligned}
\tag{5.14}
$$

Substituting by the Rhie-Chow interpolation of $\mathbf{v}$ at a face f:

$$
\begin{aligned}
& \frac{1}{\rho^1_{scale}} \frac{(\rho^1_C \alpha^1_C)^* - (\rho^1_C \alpha^1_C)^\circ}{\Delta t} V_C + \frac{1}{\rho^1_{scale}} \sum_{f\sim nb(C)} \alpha^1_f (\rho^1_f)^* \left[\overline{\mathbf{v}_f} - \overline{\mathbf{D}^{\mathbf{v}}_f} \left(\nabla p_f - \overline{\nabla p_f} \right) \right] \cdot \mathbf{S}_f \\
& + \frac{1}{\rho^1_{scale}} \sum_{f\sim nb(C)} \alpha^1_f \rho^1_f \mathbf{v}^*_f \cdot \mathbf{S}_f - \frac{1}{\rho^1_{scale}} \sum_{f\sim nb(C)} \alpha^1_f (\rho^1_f)^* \mathbf{v}^*_f \cdot \mathbf{S}_f \\
& + \sum_{k=2}^{n} \left\{ \frac{1}{\rho^k_{scale}} \frac{(\rho^k_C \alpha^k_C)^* - (\rho^k_C \alpha^k_C)^\circ}{\Delta t} V_C + \frac{1}{\rho^k_{scale}} \sum_{f\sim nb(C)} \alpha^k_f \rho^k_f \left[\overline{\mathbf{v}_f} - \overline{\mathbf{D}^{\mathbf{v}}_f} \left(\nabla p_f - \overline{\nabla p_f} \right) \right] \cdot \mathbf{S}_f \right\} \\
& = 0
\end{aligned}
\tag{5.15}
$$

Re-arranging and merging common terms for phases:

$$
\begin{aligned}
& \sum_{k=1}^{n} \left\{ \frac{1}{\rho^k_{scale}} \frac{(\rho^k_C \alpha^k_C)^* - (\rho^k_C \alpha^k_C)^\circ}{\Delta t} V_C + \frac{1}{\rho^k_{scale}} \sum_{f\sim nb(C)} \alpha^k_f \rho^k_f \left[\overline{\mathbf{v}_f} - \overline{\mathbf{D}^{\mathbf{v}}_f} \left(\nabla p_f - \overline{\nabla p_f} \right) \right] \cdot \mathbf{S}_f \right\} \\
& + \frac{1}{\rho^1_{scale}} \sum_{f\sim nb(C)} \alpha^1_f \frac{p_f}{R^1 T_f} \mathbf{v}^*_f \cdot \mathbf{S}_f - \frac{1}{\rho^1_{scale}} \sum_{f\sim nb(C)} \alpha^1_f (\rho^1_f)^* \mathbf{v}^*_f \cdot \mathbf{S}_f = 0
\end{aligned}
\tag{5.16}
$$

Doing further re-arrangements and substitutions:

$$\sum_{k=1}^{n}\left\{\frac{1}{\rho^k_{scale}}\sum_{f\sim nb(C)}\alpha^k_f\rho^k_f\left[\overline{\mathbf{v}_f}-\overline{\mathbf{D}^{\mathbf{v}}_f}\left(\nabla p_f-\overline{\nabla p_f}\right)\right]\cdot\mathbf{S}_f\right\}+\frac{1}{\rho^1_{scale}}\sum_{f\sim nb(C)}\alpha^1_f\frac{p_f}{R^1T_f}\frac{(\dot{m}^1_f)^*}{(\rho^1_f)^*}$$
$$=\frac{1}{\rho^1_{scale}}\sum_{f\sim nb(C)}\alpha^1_f(\dot{m}^1_f)^*-\sum_{k=1}^{n}\left\{\frac{1}{\rho^k_{scale}}\frac{(\rho^k_C\alpha^k_C)^*-(\rho^k_C\alpha^k_C)^\circ}{\Delta t}V_C\right\} \tag{5.17}$$

Chapter 6

Buoyancy Considerations

In cases where temperature variations are of importance, and in the presence of a gravitational field, buoyancy effects can never be ignored. In this chapter, the formulation of a free surface flow including buoyancy effects is addressed. The formulation assumes that all phases are incompressible, and the density can be formulated using the Boussinesq assumption [98]. The mathematical representation of buoyancy driven free surface flow assumes the same equations of the VOF model (equations 3.1 and 3.2), while since temperature is varying in the domain, the energy equation has to be added to the system. However, the gravitational force term in the Navier-Stokes equation has to be substituted by the Boussinesq model for density. For each phase k, the density ρ^k can be expressed as:

$$\rho^k = \rho^k_{ref}\left[1 - \beta^k(T - T_{ref})\right] \tag{6.1}$$

where β^k is the thermal expansion coefficient of phase k, and T_{ref} is the reference temperature. The mixture density ρ can then be expressed as:

$$\rho = \sum_{k=1}^{n} \alpha^k \rho^k = \sum_{k=1}^{n} \alpha^k \rho^k_{ref}\left[1 - \beta^k(T - T_{ref})\right] \tag{6.2}$$

ρ_{ref} here is the phase reference density. Then, the Navier-Stokes equation, continuity (unchanged) and energy equation is written as follows:

$$\frac{\partial \rho \mathbf{v}}{\partial t} + \nabla \cdot (\rho \mathbf{v}\mathbf{v}) = \nabla \cdot \tau - \nabla p + \sum_{k=1}^{n} \alpha^k \rho^k_{ref}\left[1 - \beta^k\left(T - T_{ref}\right)\right]\mathbf{g} \tag{6.3}$$

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \tag{6.4}$$

$$\frac{\partial \rho C_p T}{\partial t} + \nabla \cdot (\rho C_p \mathbf{v} T) = \nabla \cdot (k \nabla T) + Q^T \tag{6.5}$$

where C_p and k are respectively the mixture specific heat at constant pressure and mixture thermal conductivity of the working fluid.

Chapter 7

Boundary Conditions for Free Surface Flows

Additional discretization information at the boundary patches related to momentum and pressure equations in free surface flow modeling is presented below, while that for the volume fraction equations follows the rules of boundary conditions of a general scalar equation, and will not be presented here.

7.1 Inlet

7.1.1 Velocity Inlet

This is the boundary at which the velocity $\mathbf{v}_b$ is specified, and the normal pressure gradient is commonly set to zero (i.e. $(\nabla p)_n = 0$).

In the momentum equation, the specified velocity is simply treated explicitly as a dirichlet condition, and the momentum equation coefficients are modified as:

$$
\begin{aligned}
a_C^{uu} &:= a_C^{uu} + \mu_b^* \frac{E_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} \\
a_C^{vv} &:= a_C^{vv} + \mu_b^* \frac{E_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} \\
a_C^{ww} &:= a_C^{ww} + \mu_b^* \frac{E_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b}
\end{aligned}
$$

$$
\begin{aligned}
b_C^u &:= b_C^u + (\mu_b^* \nabla u_b^* \cdot \mathbf{T}_b) + \left(-\dot{m}_b + \mu_b^* \frac{E_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} \right) u_b \\
b_C^v &:= b_C^v + (\mu_b^* \nabla v_b^* \cdot \mathbf{T}_b) + \left(-\dot{m}_b + \mu_b^* \frac{E_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} \right) v_b \\
b_C^w &:= b_C^w + (\mu_b^* \nabla w_b^* \cdot \mathbf{T}_b) + \left(-\dot{m}_b + \mu_b^* \frac{E_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} \right) w_b
\end{aligned}
$$

And since the normal pressure gradient is set to zero, the pressure at the boundary (p_b) is extrapolated from the interior neighbouring cell (p_C) assuming a zero-order profile, such that:

$$p_b = p_C \tag{7.1}$$

so the coupling coefficients of $\mathbf{v} - p$ in the momentum equation are modified as follows:

$$\begin{aligned} a_C^{up} &:= a_C^{up} + S_b^x \\ a_C^{vp} &:= a_C^{vp} + S_b^y \\ a_C^{wp} &:= a_C^{wp} + S_b^z \end{aligned}$$

The modification of the continuity equation at a velocity inlet depends on whether any of the phases is compressible. The conditions are addressed below:

1. The situation when all of the phases are incompressible infers that the phasic mass flow rates $\dot{m}_b^k$ are known a priori (calculated as $\rho_b^k \mathbf{v}_b \cdot \mathbf{S}_b$). Thus, the coefficients of the continuity equation are modified as follows:

$$b_C^p := b_C^p - \sum_{k=1}^{n} \frac{1}{\rho_{scale}^k} (\alpha_b^k \dot{m}_b^k)^* \tag{7.2}$$

2. If any of the phases is compressible, such as the case of liquid film cooling with a weakly compressible gas, where the inlet of the coolant gas might experience weak compressibility, and assuming a zero-order profile for pressure:

$$a_C^p := a_C^p + \frac{1}{\rho_{scale}^1} (\alpha_b^1 \dot{m}_b^1)^* \frac{1}{\rho_b^1 R^1 T_b} \tag{7.3}$$

$$b_C^p := b_C^p - \sum_{k=2}^{n} \frac{1}{\rho_{scale}^k} (\alpha_b^k \dot{m}_b^k)^* \tag{7.4}$$

   where the phase 1 is the phase which corresponds to the weakly compressible gas phase, following the ideal gas law.

A final note to raise here, is that neglecting the pressure gradient at the inlet might be an inaccurate assumption in free surface flows where heavy body forces are active. Thus, a formula for the inlet pressure p_b is obtained based on Rhie-Chow interpolation. The mass flux through the inlet based on Rhie-Chow is expressed as:

$$\begin{aligned} \dot{m}_b = \rho_b \mathbf{v}_C \cdot \mathbf{S}_b - \rho_b \mathbf{D}_C^{\mathbf{v}} (\nabla p_b - \nabla p_C) \cdot \mathbf{S}_b + \rho_b \overline{\mathbf{D}_f^{\mathbf{v}}} \left(\overline{\mathbf{B}_f} - \overline{\overline{\overline{\mathbf{B}_f}}} \right) \cdot \mathbf{S}_b \\ + \rho_b \frac{a_C^{\circ} \mathbf{D}_C^{\mathbf{v}}}{V_C} \left(\mathbf{v}_b^{\circ} - \mathbf{v}_C^{\circ} \right) \cdot \mathbf{S}_b \end{aligned} \tag{7.5}$$

A formula of p_b can be extracted by re-arranging eq. 7.5:

$$\begin{aligned} p_b = p_C + \frac{d_{Cb}}{\rho_b E'_b} \Bigg[& - \dot{m}_b + \rho_b \mathbf{v}_C \cdot \mathbf{S}_b + \rho_b \nabla p_C \cdot \mathbf{S}'_b - \rho_b \nabla p_b \cdot \mathbf{T}'_b \\ & + \rho_b \overline{\mathbf{D}^{\mathbf{v}}_f} \left(\overline{\mathbf{B}_f} - \overline{\overline{\overline{\mathbf{B}_f}}} \right) \cdot \mathbf{S}_b + \rho_b \frac{a^{\circ}_C \mathbf{D}^{\mathbf{v}}_C}{V_C} \left(\mathbf{v}^{\circ}_b - \mathbf{v}^{\circ}_C \right) \cdot \mathbf{S}_b \Bigg] \end{aligned} \tag{7.6}$$

Putting in a more compact form:

$$p_b = p_C + C_1 \mathbf{v}_C \cdot \mathbf{S}_b + C_2 \tag{7.7}$$

where

$$C_1 = \frac{d_{Cb}}{E'_b}$$

$$C_2 = C_1 \left[-\frac{\dot{m}_b}{\rho_b} + \nabla p_C \cdot \mathbf{S}'_b - \nabla p_b \cdot \mathbf{T}'_b + \overline{\mathbf{D}^{\mathbf{v}}_f} \left(\overline{\mathbf{B}_f} - \overline{\overline{\overline{\mathbf{B}_f}}} \right) \cdot \mathbf{S}_b + \frac{a^{\circ}_C \mathbf{D}^{\mathbf{v}}_C}{V_C} \left(\mathbf{v}^{\circ}_b - \mathbf{v}^{\circ}_C \right) \cdot \mathbf{S}_b \right]$$

7.1.2 Pressure Inlet

The static pressure at the boundary is determined from the total pressure definition:

$$p_b = p_0 - \frac{1}{2} \rho_b \|\mathbf{v}_b\|^2 \tag{7.8}$$

while the velocity is determined from the mass flux:

$$\mathbf{v}_b = \frac{\dot{m}_b}{\rho_b \|\mathbf{S}_b\|} \mathbf{e}^{\mathbf{v}} \tag{7.9}$$

where $\mathbf{e}^{\mathbf{v}}$ is a specified direction of the velocity. $\dot{m}_b$ is determined based on Rhie-Chow interpolation:

$$\dot{m}_b = \rho_b \mathbf{v}_C \cdot \mathbf{S}_b - \rho_b \mathbf{D}^{\mathbf{v}}_b (\nabla p_b - \nabla p_C) \cdot \mathbf{S}_b \tag{7.10}$$

The momentum and continuity equations are modified accordingly.

7.2 Outlet

7.2.1 Pressure Outlet

The commonly used boundary condition at the outlet is the pressure outlet boundary condition, at which the static pressure is specified and the velocity vector is extrapolated from interior. The coefficients of the momentum equation are modified as follows:

$$\begin{aligned} a^{uu}_C &:= a^{uu}_C + \dot{m}^*_b \\ a^{vv}_C &= a^{vv}_C + \dot{m}^*_b \\ a^{ww}_C &:= a^{ww}_C + \dot{m}^*_b \end{aligned}$$

$$
\begin{aligned}
b_C^u &:= b_C^u - p_b^* S_b^x \\
b_C^v &:= b_C^v - p_b^* S_b^y \\
b_C^w &:= b_C^w - p_b^* S_b^z
\end{aligned}
$$

On another hand, the coefficients of the continuity equation become as follows:

$$
\begin{aligned}
a_C^{pu} &:= a_C^{pu} + \sum_{k=1}^{n} \frac{1}{\rho_{scale}^k} (\alpha_b^k \rho_b^k)^* S_b^x \\
a_C^{pv} &:= a_C^{pv} + \sum_{k=1}^{n} \frac{1}{\rho_{scale}^k} (\alpha_b^k \rho_b^k)^* S_b^y \\
a_C^{pw} &:= a_C^{pw} + \sum_{k=1}^{n} \frac{1}{\rho_{scale}^k} (\alpha_b^k \rho_b^k)^* S_b^z
\end{aligned}
$$

$$
a_C^{pp} := a_C^{pp} + \sum_{k=1}^{n} \frac{1}{\rho_{scale}^k} (\alpha_b^k \rho_b^k)^* \mathcal{D}_b
$$

$$
b_C^p := b_C^p + \sum_{k=1}^{n} \frac{1}{\rho_{scale}^k} \left[-(\alpha_b^k \rho_b^k)^* (\mathbf{D}_b^{\mathbf{v}} \nabla p_C^* \cdot \mathbf{S}_b) + (\alpha_b^k \rho_b^k)^* \mathcal{D}_b p_b \right]
$$

A point worth mentioning here is flow reversal at outlet. For many cases, recirculation near the outlet takes place, and in order to conserve mass, the corresponding part of the outlet experiences a reversed flow. While it is physically legitimate to allow flow reversal, it is not the case at a pressure outlet. A different boundary condition called ”opening” is designed to allow flow reversal and it is addressed in a later section. For a pressure outlet, if any flow reversal at any face of the boundary is detected, it is blocked by placing a virtual wall at that face. The influence of the coefficients of the momentum and continuity equations at these specific faces with virtual walls follow the same discretization principles of a no-slip wall, and this is to be discussed in the section of wall boundary condition.

7.2.2 Outflow

This boundary condition is appropriate when no information about any of the fields is available, so it assumes a zero-gradient boundary condition for all the fields across the boundary. This boundary condition is usually not stable, and has to be properly used in arbitrary cases such as cases which involve supersonic flows where disturbances may only travel downstream. Other cases, more related to the context of free surface flows, are open channel cases, where for a supercritical flow ($Fr > 1$), disturbances in the channel will travel downstream, hence, the flow fields at the outlet are always extrapolated from interior neighbouring cells; this is the core concept of an outflow boundary condition. The modifications in the momentum equation coefficients are:

$$\begin{aligned} a_C^{uu} &:= a_C^{uu} + \dot{m}_b^* \\ a_C^{vv} &= a_C^{vv} + \dot{m}_b^* \\ a_C^{ww} &:= a_C^{ww} + \dot{m}_b^* \end{aligned}$$

$$\begin{aligned} a_C^{up} &:= a_C^{up} + S_b^x \\ a_C^{vp} &:= a_C^{vp} + S_b^y \\ a_C^{wp} &:= a_C^{wp} + S_b^z \end{aligned}$$

The modification of the continuity equation for the two cases, all phases are incompressible, or a single compressible phase:

1. The situation when all of the phases are incompressible:

$$b_C^p := b_C^p - \sum_{k=1}^{n} \frac{1}{\rho_{scale}^k} (\alpha_b^k \dot{m}_b^k)^* \tag{7.11}$$

2. If one of the phases is compressible:

$$a_C^p := a_C^p + \frac{1}{\rho_{scale}^1} (\alpha_b^1 \dot{m}_b^1)^* \frac{1}{\rho_b^1 R^1 T_b} \tag{7.12}$$

$$b_C^p := b_C^p - \sum_{k=2}^{n} \frac{1}{\rho_{scale}^k} (\alpha_b^k \dot{m}_b^k)^* \tag{7.13}$$

In the case of flow reversal, the same treatment as in the pressure outlet is made.

7.3 Symmetry

There's no mass flux entering or leaving a symmetry plane. Any scalar quantity (i.e. pressure) can be simply extrapolated from the interior, however, the velocity vector at a symmetry plane is equal to the parallel component of the velocity vector in the neighbouring element, which means that there exists a normal stress on the fluid at the symmetry. This normal stress influences the momentum equation by means of interaction between velocity components, such as shown below:

$$a_C^{uv} := a_C^{uv} + 2\frac{\mu_b^* S_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} n_b^x n_b^y \qquad a_C^{vu} := a_C^{vu} + 2\frac{\mu_b^* S_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} n_b^y n_b^x$$
$$a_C^{uw} := a_C^{uw} + 2\frac{\mu_b^* S_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} n_b^x n_b^z \qquad a_C^{wu} := a_C^{wu} + 2\frac{\mu_b^* S_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} n_b^z n_b^x$$
$$a_C^{vw} := a_C^{vw} + 2\frac{\mu_b^* S_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} n_b^y n_b^z \qquad a_C^{wv} := a_C^{wv} + 2\frac{\mu_b^* S_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} n_b^z n_b^y$$

$$a_C^{up} := a_C^{up} + S_b^x$$
$$a_C^{vp} := a_C^{vp} + S_b^y$$
$$a_C^{wp} := a_C^{wp} + S_b^z$$

No modification to the continuity equation at a symmetry plane because the mass flux across it is zero.

7.4 Wall

At a wall, the mass flux is zero, and the pressure is reasonably set to be zero-gradient across it. The velocity vector assumes the velocity of the wall in the case of no-slip condition, while for a slip wall, the situation is exactly the same as a symmetry plane and the influence on the equations is the same. Referring back to no-slip condition, the stress which evolves at a no-slip wall influences the momentum equation coefficients by means of interactions between velocity components (not to be confused with those at symmetry plane); they are presented here:

$$a_C^{uv} := a_C^{uv} - \frac{\mu_b^* S_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} n_b^x n_b^y \qquad a_C^{vu} := a_C^{vu} - \frac{\mu_b^* S_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} n_b^y n_b^x$$
$$a_C^{uw} := a_C^{uw} - \frac{\mu_b^* S_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} n_b^x n_b^z \qquad a_C^{wu} := a_C^{wu} - \frac{\mu_b^* S_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} n_b^z n_b^x$$
$$a_C^{vw} := a_C^{vw} - \frac{\mu_b^* S_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} n_b^y n_b^z \qquad a_C^{wv} := a_C^{wv} - \frac{\mu_b^* S_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} n_b^z n_b^y$$

$$a_C^{up} := a_C^{up} + S_b^x$$
$$a_C^{vp} := a_C^{vp} + S_b^y$$
$$a_C^{wp} := a_C^{wp} + S_b^z$$

On another hand, since free surface flows are necessarily simulated with body forces like gravitational acceleration and/or surface tension, the pressure gradient at wall can not be ignored [99]. A Rhie-Chow interpolation may be used to formulate a high-order extrapolation method for pressure at the wall. At a boundary face b, the mass flux is:

$$\begin{aligned}\dot{m}_b = \rho_b \mathbf{v}_C \cdot \mathbf{S}_b - \rho_b \mathbf{D}^{\mathbf{v}}_C (\nabla p_b - \nabla p_C) \cdot \mathbf{S}_b + \rho_b \overline{\mathbf{D}^{\mathbf{v}}_f} \left(\mathbf{B}_C - \overline{\overline{\mathbf{B}_C}}\right) \cdot \mathbf{S}_b \\ + \rho_b \frac{a^{\circ}_C \mathbf{D}^{\mathbf{v}}_C}{V_C} \left(\mathbf{v}^{\circ}_b - \mathbf{v}^{\circ}_C\right) \cdot \mathbf{S}_b\end{aligned} \tag{7.14}$$

For a no-slip wall, $\dot{m}_b = 0$, so a formula of p_b can be extracted by re-arranging eq. 7.14:

$$\begin{aligned}p_b = p_C + \frac{d_{Cb}}{\rho_b E'_b} \Big[\rho_b \mathbf{v}_C \cdot \mathbf{S}_b + \rho_b \nabla p_C \cdot \mathbf{S}'_b - \rho_b \nabla p_b \cdot \mathbf{T}'_b \\ + \rho_b \overline{\mathbf{D}^{\mathbf{v}}_f} \left(\mathbf{B}_C - \overline{\overline{\mathbf{B}_C}}\right) \cdot \mathbf{S}_b + \rho_b \frac{a^{\circ}_C \mathbf{D}^{\mathbf{v}}_C}{V_C} \left(\mathbf{v}^{\circ}_b - \mathbf{v}^{\circ}_C\right) \cdot \mathbf{S}_b \Big]\end{aligned} \tag{7.15}$$

Putting in a more compact form:

$$p_b = p_C + C_1 \mathbf{v}_C \cdot \mathbf{S}_b + C_2 \tag{7.16}$$

where

$$\begin{aligned}C_1 &= \frac{d_{Cb}}{E'_b} \\ C_2 &= C_1 \left[\nabla p_C \cdot \mathbf{S}'_b - \nabla p_b \cdot \mathbf{T}'_b + \overline{\mathbf{D}^{\mathbf{v}}_f} \left(\mathbf{B}_C - \overline{\overline{\mathbf{B}_C}}\right) \cdot \mathbf{S}_b + \frac{a^{\circ}_C \mathbf{D}^{\mathbf{v}}_C}{V_C} \left(\mathbf{v}^{\circ}_b - \mathbf{v}^{\circ}_C\right) \cdot \mathbf{S}_b \right]\end{aligned}$$

The modifications of the momentum equation coefficients are the same as those for the standard no-slip wall boundary condition, with an additional modification to the $\mathbf{v} - p$ coefficients as well as the rhs coefficients:

$$\begin{aligned}a^{uu}_C &:= a^{uu}_C + \mu^*_b \frac{E_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} + C_1 (S^x_b)^2 \\ a^{vv}_C &:= a^{vv}_C + \mu^*_b \frac{E_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} + C_1 (S^y_b)^2 \\ a^{ww}_C &:= a^{ww}_C + \mu^*_b \frac{E_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} + C_1 (S^z_b)^2\end{aligned}$$

$$\begin{aligned}a^{uv}_C &:= a^{uv}_C - \frac{\mu^*_b S_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} n^x_b n^y_b + C_1 S^y_b & \quad a^{vu}_C &:= a^{vu}_C - \frac{\mu^*_b S_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} n^y_b n^x_b + C_1 S^x_b \\ a^{uw}_C &:= a^{uw}_C - \frac{\mu^*_b S_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} n^x_b n^z_b + C_1 S^z_b & \quad a^{wu}_C &:= a^{wu}_C - \frac{\mu^*_b S_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} n^z_b n^x_b + C_1 S^x_b \\ a^{vw}_C &:= a^{vw}_C - \frac{\mu^*_b S_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} n^y_b n^z_b + C_1 S^z_b & \quad a^{wv}_C &:= a^{wv}_C - \frac{\mu^*_b S_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} n^z_b n^y_b + C_1 S^y_b\end{aligned}$$

$$
\begin{aligned}
a_C^{up} &:= a_C^{up} + S_b^x \\
a_C^{vp} &:= a_C^{vp} + S_b^y \\
a_C^{wp} &:= a_C^{wp} + S_b^z
\end{aligned}
$$

$$
\begin{aligned}
b_C^u &:= b_C^u + \mu_b^* \nabla u_b^* \cdot \mathbf{T}_b + \left(\mu_b^* \nabla \mathbf{v}_b^T \cdot \mathbf{S}_b\right) \cdot \hat{\mathbf{i}} + \mu_b (\nabla \cdot \mathbf{v})_b S_b^x - C_2 S_b^x \\
b_C^v &:= b_C^v + \mu_b^* \nabla v_b^* \cdot \mathbf{T}_b + \left(\mu_b^* \nabla \mathbf{v}_b^T \cdot \mathbf{S}_b\right) \cdot \hat{\mathbf{j}} + \mu_b (\nabla \cdot \mathbf{v})_b S_b^y - C_2 S_b^y \\
b_C^w &:= b_C^w + \mu_b^* \nabla w_b^* \cdot \mathbf{T}_b + \left(\mu_b^* \nabla \mathbf{v}_b^T \cdot \mathbf{S}_b\right) \cdot \hat{\mathbf{k}} + \mu_b (\nabla \cdot \mathbf{v})_b S_b^z - C_2 S_b^z
\end{aligned}
$$

And as mentioned previously, there's no mass flux through the wall, which means that the continuity equation coefficients are not modified.

7.5 Opening

The opening boundary condition combines the properties of an inlet and an outlet. In fact, it is a pressure outlet boundary condition with the specialty of allowing flow reversal, or what is called "back-flow". So the same discretzation rules apply as in the pressure outlet boundary condition if the flow is exiting the boundary, while for the back-flow which is entering, a special treatment is made based on the assumption that the total pressure p_0 is fixed. The following determines the influence on the momentum and continuity equations. Assuming the case where all phases are incompressible, p_b, the static pressure at the boundary, is defined as:

$$
p_b = p_0 - \frac{1}{2} \rho_b \|\mathbf{v}_b\|^2 \tag{7.17}
$$

p_b could be absolute or relative depending on the definition of the problem, and ρ_b here is the mixture density at the boundary. Knowing that some cases which include a gas could experience only weak compressibility, the pressure at the boundary p_b is assumed to follow the same formula above. The momentum equation coefficients are modified as follows:

$$
\begin{aligned}
a_C^{uu} &:= a_C^{uu} + \mu_b^* \frac{E_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} \\
a_C^{vv} &:= a_C^{vv} + \mu_b^* \frac{E_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b} \\
a_C^{ww} &:= a_C^{ww} + \mu_b^* \frac{E_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b}
\end{aligned}
$$

$$b_C^u := b_C^u + (\mu_b^* \nabla u_b^* \cdot \mathbf{T}_b) + \left(-\dot{m}_b + \mu_b^* \frac{E_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b}\right) u_b^* - \left(p_0 - \frac{1}{2}\rho_b \|\mathbf{v}_b^*\|^2\right) S_b^x$$

$$b_C^v := b_C^v + (\mu_b^* \nabla v_b^* \cdot \mathbf{T}_b) + \left(-\dot{m}_b + \mu_b^* \frac{E_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b}\right) v_b^* - \left(p_0 - \frac{1}{2}\rho_b \|\mathbf{v}_b^*\|^2\right) S_b^y$$

$$b_C^w := b_C^w + (\mu_b^* \nabla w_b^* \cdot \mathbf{T}_b) + \left(-\dot{m}_b + \mu_b^* \frac{E_b}{\mathbf{d}_{Cb} \cdot \mathbf{n}_b}\right) w_b^* - \left(p_0 - \frac{1}{2}\rho_b \|\mathbf{v}_b^*\|^2\right) S_b^z$$

The modification of the continuity equation at an opening face where flow is reversed is treated here in the same way the continuity equation is treated at a velocity inlet:

1. If all phases are incompressible:

$$b_C^p := b_C^p - \sum_{k=1}^{n} \frac{1}{\rho_{scale}^k} (\alpha_b^k \dot{m}_b^k)^* \tag{7.18}$$

2. If one of the phases is compressible:

$$a_C^p := a_C^p + \frac{1}{\rho_{scale}^1} (\alpha_b^1 \dot{m}_b^1)^* \frac{1}{\rho_b^1 R^1 T_b} \tag{7.19}$$

$$b_C^p := b_C^p - \sum_{k=2}^{n} \frac{1}{\rho_{scale}^k} (\alpha_b^k \dot{m}_b^k)^* \tag{7.20}$$

   where the phase 1 is the phase which corresponds to the weakly compressible gas phase, following the ideal gas law.

7.6 Cyclic Arbitrary Mesh Interface (AMI)

This is an example of the so-called coupled boundaries, and it is also called periodic boundary condition. Each cyclic AMI boundary is a part of a pair of boundaries, one of them is called master boundary and the other is called slave boundary. In its essence, this coupled boundary is an internal interface, such that its faces are treated just like the other internal faces of the mesh. However, an additional treatment has to be made here related to the transformation of the velocity vector or any other vectorial quantity depending on the type of the boundary whether translational or rotational. If translational, the transformation is simply the identity matrix, while if rotational, a convenient transformation matrix is defined based on the geometry of the surfaces.

Chapter 8

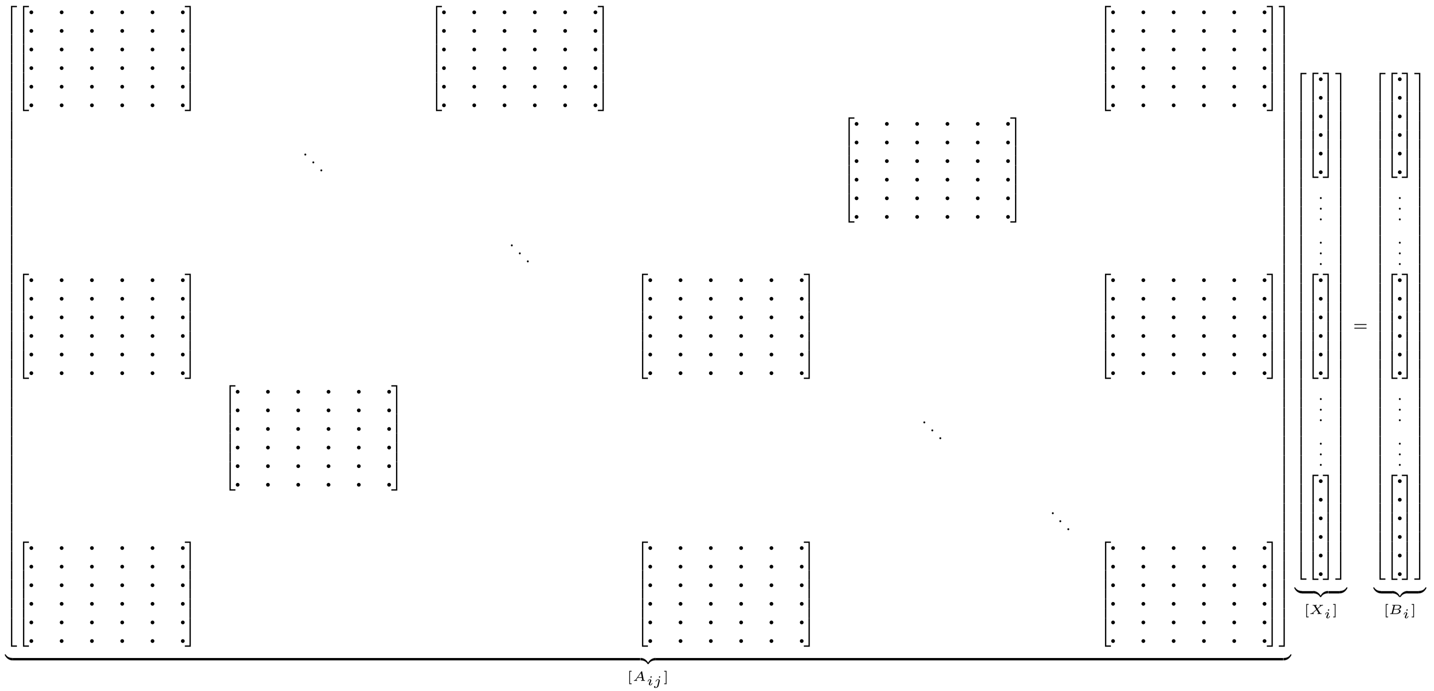
Linear Solver Theory

8.1 The Discrete Linear System of the Coupled Solver

Besides an appropriate fully implicit discretization of the Navier–Stokes' equations, continuity and volume fraction equations, the performance of the coupled algorithm is critically dependent on the proper implementation of an iterative solver to ensure that the increase in computational time incurred in the solution of the enlarged system of equations does not counter balance the advantage of the higher convergence rate. In one-component systems, the coefficients represent the influences between neighboring elements, i.e. spatial influences. For a coupled system, in addition to the spatial connectivity, inter-component connections arise. This renders the use of the algebraic multigrid iterative solver described above unsuitable. To circumvent this hurdle and efficiently employ the one-component algebraic multigrid algorithm to solving the coupled system, the original spatial connectivity array describing the topology of the mesh is retained for use in the agglomeration procedure of the multigrid algorithm, while an expanded connectivity array that accounts for the inter-variable influences is constructed for the iterative solver. Drawish et. el [63] has described an efficient implementation of their coupled solver. To this end, an overview of the construction of the discrete algebraic coupled VOF matrix is presented. Starting from the discrete algebraic system presented earlier in eq. 4.55 for an element C in the computational domain constituting N elements, re-stated here in compact form:

$$A^{\bullet}_{ij} X^{\bullet}_{i} + \sum_{F \sim NB(C)} A^{\circ}_{ij} X^{\circ}_{i} = B_i \tag{8.1}$$

where $A^{\bullet}_{ij}$ and $X^{\bullet}_{i}$ are the central coefficient block and its corresponding solution vector, while A°_{ij} and X°_{i} are the extra-diagonal coefficient block and its corresponding solution vector. B_i is the right-hand-side of the equation. Eq. 8.1 is assembled into a global matrix, a matrix which is in fact a sparse block matrix where the iterative solution has to be of block-type. The system involving the global matrix $[A_{ij}]$ with the solution vector $[X_i]$ and the right-hand-side $[B_i]$ has the form presented in the following (assuming a two-phase flow so 6×6 blocks are involved).



(8.2)

The global matrix in the system above is a sparse matrix, such that most of its extra-diagonal entries are zeros. The fact that it is a sparse matrix necessitates the storage of non-zero entries only within proper containers.

8.2 Linear Solver

Large discrete systems may be solved using many different methods, some of which are direct and others are indirect iterative. Direct methods are not suitable for very large systems due to the fact that they require enormous memory storage capabilities as compared to iterative methods, and are also very time consuming. In addition to that, the more non-linear the problem is, the more time consuming it would be, which is very prominent in fluid flows. In this work, a combination of ILU(0) [100] algorithm with a corrective multigrid scheme is used [101]. Not only does this combined method provide a low storage feature, but it also has a high convergence rate due to the implementation of the multigrid algorithm.

8.2.1 Incomplete LU (ILU) Decomposition

The system to be solved is of the form

$$[A_{ij}][X_i] = [B_i] \tag{8.3}$$

The global matrix $[A_{ij}]$ can be decomposed into incomplete lower and upper matrices in order to keep same sparsity structure as that of $[A_{ij}]$

$$[A_{ij}] = [\overline{L}_{ij}][\overline{U}_{ij}] + [R_{ij}] \tag{8.4}$$

where $[R_{ij}]$ is the residual of factorization procedure. To avoid storing additional nonzero locations required by $[R_{ij}]$, the matrix $[A_{ij}]$ is approximated as

$$[A_{ij}] \approx [\overline{L}_{ij}][\overline{U}_{ij}] \tag{8.5}$$

This is referred to as ILU(0) decomposition, and it is an incomplete factorization based on taking a subset $[P_{ij}]$ of nonzero elements from the global coefficient matrix $[A_{ij}]$ [100]. It does not produce any non-zero element outside the sparsity of the original matrix $[A_{ij}]$, thus, the iterative solver requires by far as much storage as $[A_{ij}]$. The factorization algorithm is presented in algorithm 1 for determining approximations for the incomplete lower and upper triangular matrices $[\overline{L}_{ij}]$ and $[\overline{U}_{ij}]$ respectively.

The solution process starts with reformulating 8.3 in correction form in such a way to be able to apply an iterative process. Without further details the system appears as

$$[\overline{L}_{ij}][\overline{U}_{ij}][X'_i]^{(n)} = [B_i] - [A_{ij}][X_i]^{(n-1)} \tag{8.6}$$

Algorithm 1 ILU(0) factorization algorithm

```
for $k = 1$ to $N - 1$ do
    for $k = 1$ to $N - 1$ do
        if $A_{ik} \neq 0$ then
            $A_{ik} = A_{ik} \times A_{kk}^{-1}$ (lower triangle values)
            for $k = 1$ to $N - 1$ do
                if $A_{ij} \neq 0$ then
                    $A_{ij} = A_{ij} - A_{ik} \times A_{kj}$ (upper triangle values)
                end if
            end for
        end if
    end for
end for
```

8.2.2 ILU(0) Factorization Preconditioner

This iterative solver, in its aforementioned version, would deteriorate the convergence rate. To remedy this drawback, the iterative solver is used as a preconditioner of an algebraic multigrid solver. And since the incomplete factorization mimics the nonzero elements sparsity of the original matrix such that the preconditioner has exactly the size of the original matrix, it is possible to write the preconditioner in the form

$$[P_{ij}] = ([D^*_{ij}] + [L_{ij}])[D^*_{ij}]^{-1}([D^*_{ij}] + [U_{ij}]) = [\overline{L}_{ij}][\overline{U}_{ij}] \tag{8.7}$$

The algorithm to determine the $[D^*_{ij}]$ matrix is presented in algorithm 2.

Algorithm 2 Determination of $[D^*_{ij}]$

```
for $k = 1$ to $N$ do
    $D^*_{ii} = A_{ii}$
end for
for $i = 1$ to $N$ do
    for $j = i + 1$ to $N$ do
        if $A_{ij} \neq 0$ and $A_{ji} \neq 0$ then
            $D^*_{jj} = D^*_{jj} - (A_{ji} \times {D^*}_{ii}^{-1})A_{ij}$
        end if
    end for
end for
```

The forward and backward solution algorithm with the mentioned ILU method is presented in algorithm 3

Algorithm 3 Solution algorithm

```
for $i = 1$ to $N$ do
    for $j = 1$ to $i - 1$ do
        $T_i = D_{ii}^{*-1}(R_i - L_{ij}T_j)$
    end for
end for
for $i = N$ to $1$ do
    for $j = i + 1$ to $N$ do
        $X_i' = T_i - D_{ii}^{*-1}(U_{ij}T_j)$
    end for
end for
```

8.2.3 Overview of the Multigrid Solver

The basic concept behind multigrid methods is to accelerate the convergence of an iterative method, which typically reduces the high frequency errors in the system, through a correction scheme to the finest grid, achieved by solving a coarser grid. It is much cheaper to solve the coarse problem, which is similar to the fine one in that it also has low and high frequency errors. This coarse grid can similarly be solved by any iterative method (called also relaxation sweeps) and by the aid of further coarse grids. Therefore, this is a recursive process which is repeated until the latest generated coarse grid can be solved through a direct solver with a negligible computational cost relative to the cost of a single relaxation sweep on the finest grid. When this recursive method finishes a cycle, it should typically reduce errors of all frequencies by a fixed factor well below unity [102]. To sum up, multigrid techniques are said to be one of the fastest acceleration techniques. Mutligrid methods were first proposed by Federenko [103] and Poussin [104], and have later gained popularity after the unique work of Brandt [105]. All segregated and coupled results presented in this work are generated using an algebraic multigrid with an ILU(0) solver as a smoother.

Chapter 9

Technique Assessment

In the current section, a set of technique assessment tests are presented to validate the effectiveness of the different techniques implemented in the coupled VOF solver.

9.1 Verification of Force Balancing

The force-balancing methods are validated here for a set of simple free surface flow problems involving equilibrium conditions; this is the most suitable way for assessing the level of force balancing in the presented coupled VOF solver. It is noted that all of the methods are supposed to have a positive influence on the stability of the interface as compared to a non-balanced situation, but with different levels of stability and robustness.

9.1.1 Simple Grid Analysis for Water/Air Medium at Equilibrium

Figure 9.1 shows 3 different schematics for water/air phases at equilibrium. The grids are respectively a simple cartesian, simple structured non-orthogonal quadrilateral and simple triangular. The gray-colored elements are those filled with water, while the white-colored elements are filled with air. Elements with a bright-gray color are elements sitting at the interface. The thick-bordered region in the cases consists of the elements and faces that are considered for evaluation; the selected elements will represent all the ranges of evaluation quantities. This section aims at highlighting the level of force-balancing of each of the mentioned methods.

Specific quantities have to be speculated. Method 1 states that the pressure interpolation to the face, required in the Gauss formula for determining the pressure gradient, is to be made using a harmonic interpolation. The quantity to observe here is the difference between the discrete pressure gradient and body force terms. Denoting the difference between the two terms by Δ_b, as shown in the equation

$$\Delta_b = \|\nabla p - \mathbf{B}_C\| \tag{9.1}$$

Δ_b is also called force-balance defect. Method 2 involves modifying the pressure field whose momentum equation is reported previously in eq. 4.69. The pressure gradient and the source term (last term) in eq. 4.69 have to be equal to maintain equilibrium (this is not expected at an interface as mentioned previously). The defect for this method is defined as

$$\Delta_b = \|\nabla p' + \mathbf{g} \cdot (\mathbf{R} - \mathbf{R}_{ref})\nabla \rho\| \tag{9.2}$$

On another hand, the methods which involve a reconstructed body force (methods 3a, 3b and 3c) will be speculated as follows. Starting with the momentum equation 3.1 with a gravitational body force $\mathbf{B} = (\rho - \rho_0)\mathbf{g}$, the discrete body force over an arbitrary element C, $\mathbf{B}_C$, is replaced by a reconstructed body force $\overline{\overline{\mathbf{B}_C}}$. The force-balance defect Δ_b is defined as

$$\Delta_b = \|\nabla p - \overline{\overline{\mathbf{B}_C}}\| \tag{9.3}$$

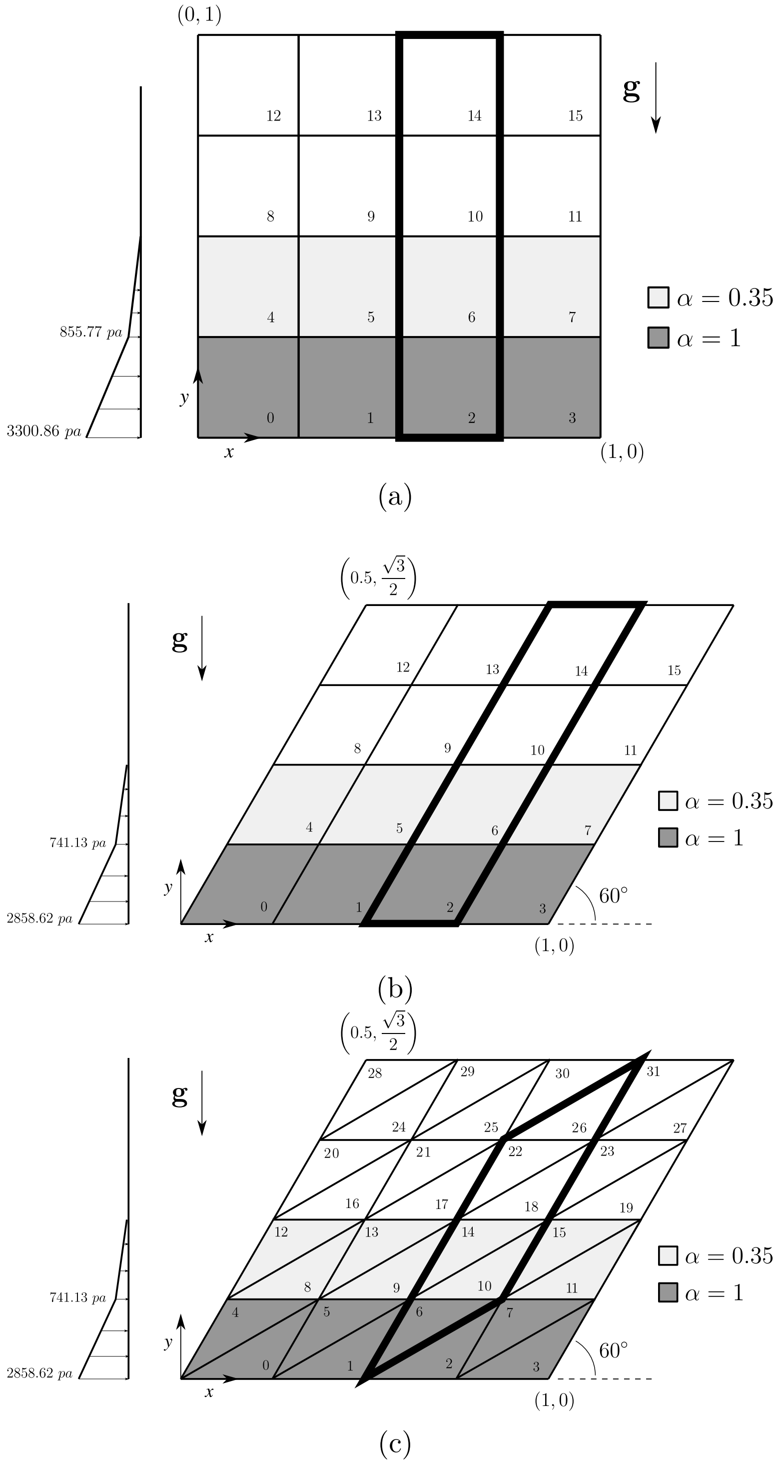
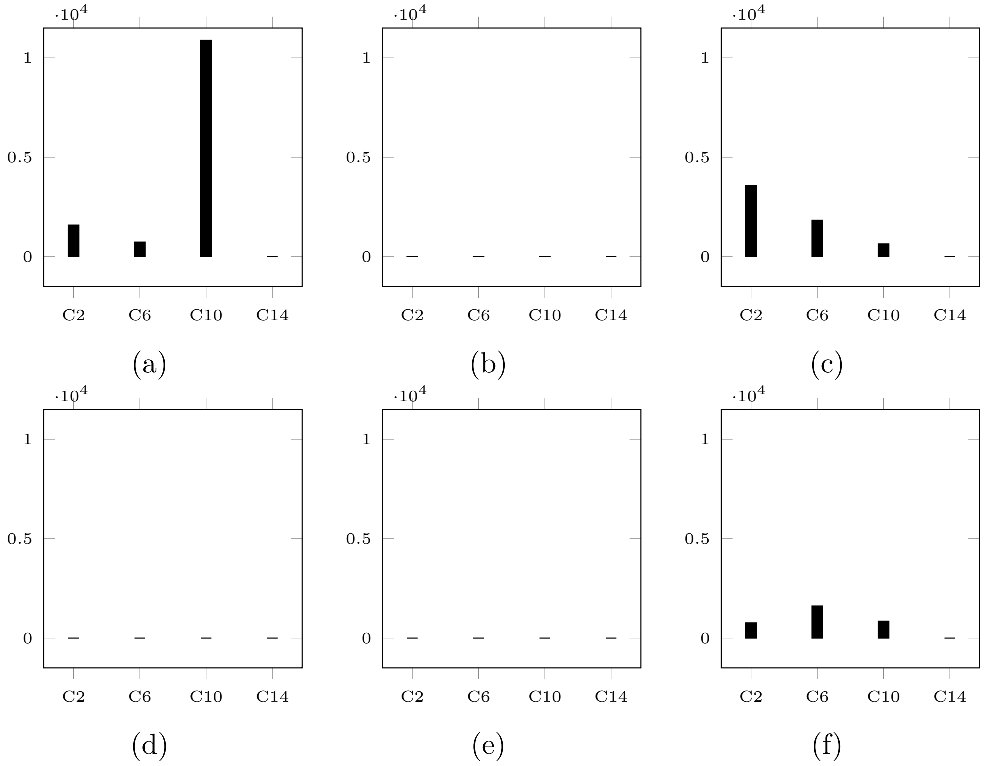


Figure 9.1: Schematics of the water-air phases at equilibrium test cases for meshes of type: (a) simple cartesian, (b) simple quadrilateral, and (c) simple triangular. *Dimensions in m*

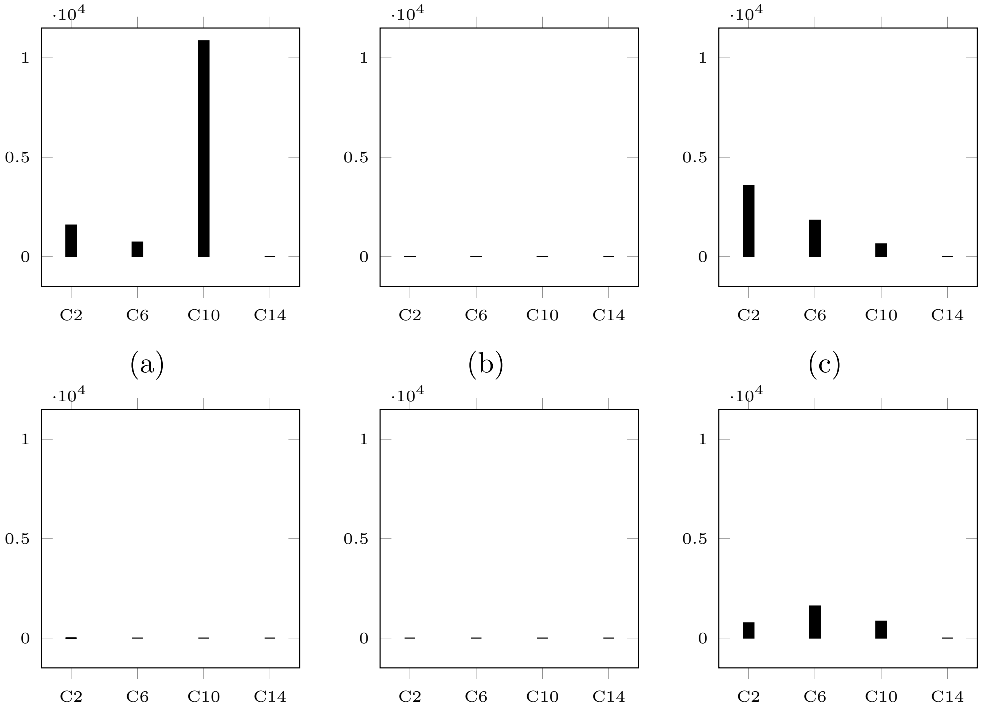
Figures 9.2, 9.3 and 9.4 show the results corresponding to the force-balance defect. For the simple cartesian mesh (figure 9.2), the defect when no balancing strategy is used shows very large values, comes after in defect levels, are the shifted pressure formulation especially at the interface elements, then the combined traits method where also some levels of defect are observed. In case 2 which involves a simple non-orthogonal mesh (figure 9.3), the defect levels are pretty much like those in case 1. For case 3 where the mesh is triangular, the method that does not include balancing, the shifted pressure formulation and the combined traits method have shown high levels of defect, with higher levels being associated with the shifted pressure formulation. This study is a preliminary test for the methods only, designed to highlight the level at which the force-balancing method is ensuring balance in terms.



(a) (b) (c)

(d) (e) (f)

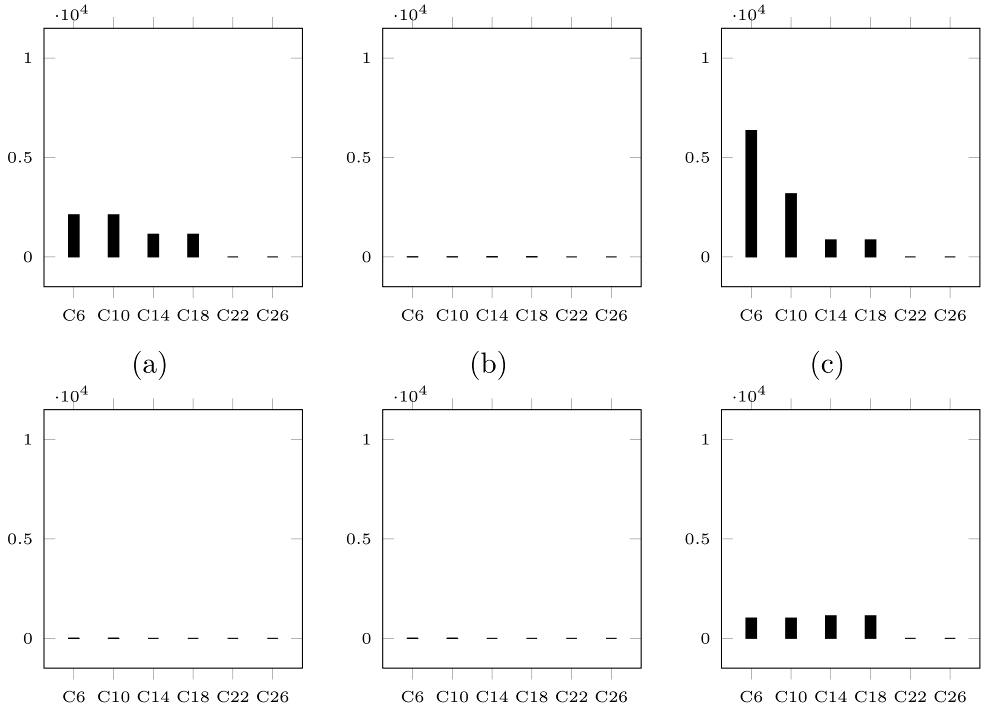
Figure 9.2: Force-balance defect for case 1 with (a) no balance strategy, (b) harmonic pressure interpolation, (c) shifted pressure formulation, (d) reconstruction with linear interpolation for pressure and body force, (e) reconstruction with harmonic interpolation for pressure and body force, and (f) reconstruction with linear interpolation for pressure and harmonic for body force



(a) (b) (c)

(d) (e) (f)

Figure 9.3: Force-balance defect for case 2 with (a) no balance strategy, (b) harmonic pressure interpolation, (c) shifted pressure formulation, (d) reconstruction with linear interpolation for pressure and body force, (e) reconstruction with harmonic interpolation for pressure and body force, and (f) reconstruction with linear interpolation for pressure and harmonic for body force



(a) (b) (c)

(d) (e) (f)

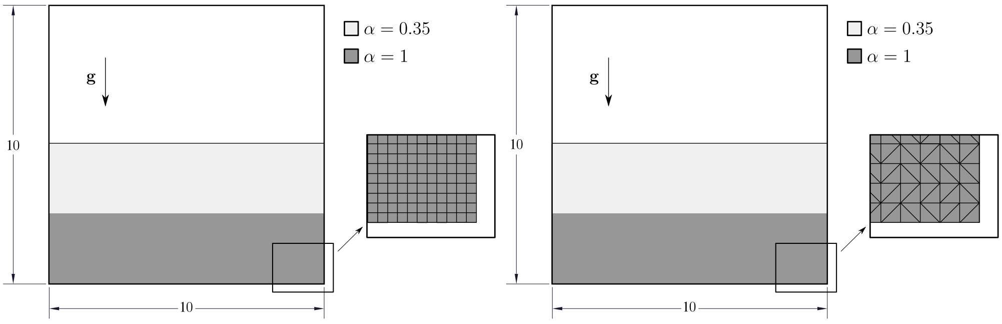
Figure 9.4: Force-balance defect for case 3 with (a) no balance strategy, (b) harmonic pressure interpolation, (c) shifted pressure formulation, (d) reconstruction with linear interpolation for pressure and body force, (e) reconstruction with harmonic interpolation for pressure and body force, and (f) reconstruction with linear interpolation for pressure and harmonic for body force

9.1.2 Water/Air Medium at Equilibrium

A full-scale run will be made in this section for cases 1 and 2 (see figure 9.5) using the aforementioned force balancing methods. Case 1 includes 46 × 46 uniform elements while case 2 contains 2048 triangular elements (roughly equivalent to 46 × 46). Steady-state simulations for the 2 cases are made using the VOF solver. Along a vertical line from the ground to the top, some quantities are plotted, and they include the pressure distribution p *vs.* y, pressure gradient magnitude $\|\nabla p\|$ *vs.* y, and velocity magnitude $\|\mathbf{v}\|$ *vs.* y. Since method 2 involves a shifted pressure, the real pressure (∇p) has to be calculated back from the shifted pressure gradient ($\nabla p'$) according to

$$\nabla p = \nabla p' + (\rho - \rho_0)\mathbf{g} + \mathbf{g} \cdot (\mathbf{R} - \mathbf{R}_{ref})\nabla \rho \tag{9.4}$$

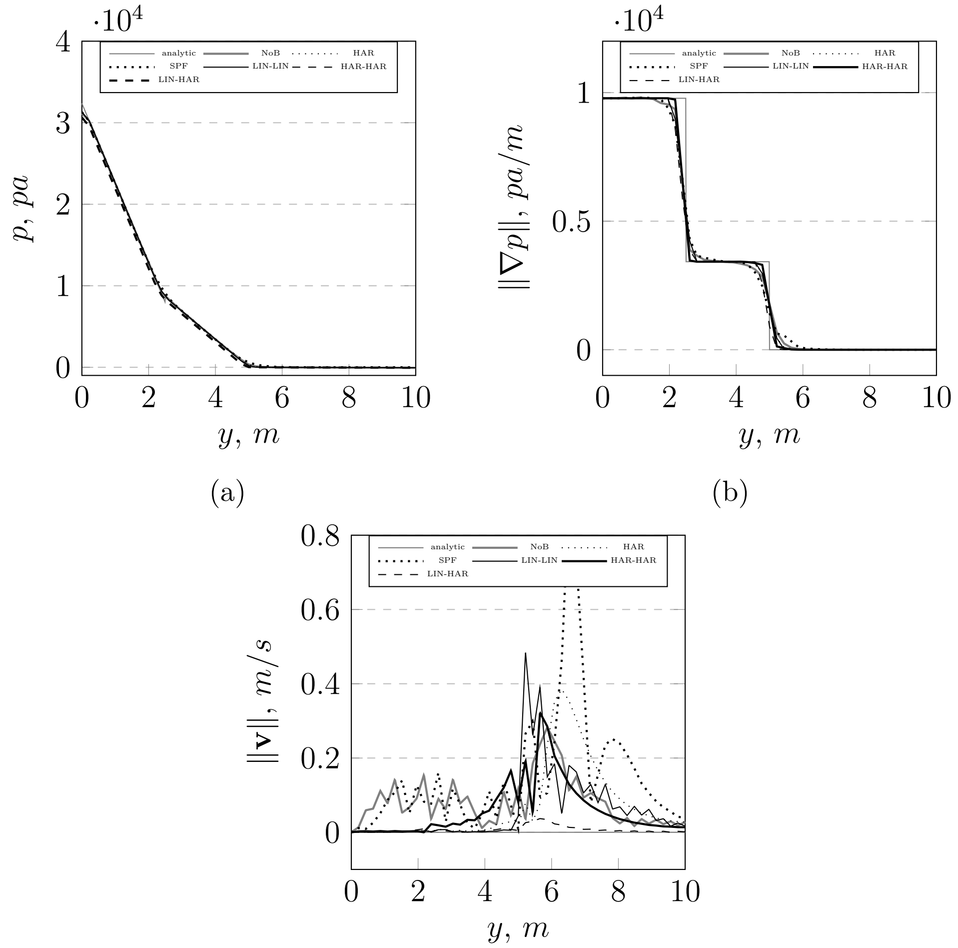
Figures 9.6 and 9.7 show the various results for the two meshes. For the cartesian mesh, it is obvious that all the methods have produced pressure distributions that are consistent with the analytic solution, while for the pressure gradient, though they only showed small defect compared to the analytically calculated pressure gradient magnitude, the shifted pressure formulation showed more explicit defect than the other methods. As for the velocity magnitude distribution, the methods generally showed unstable fluctuations, especially near the interface ($y \sim 0.5[m]$), with the combined traits method (linear for pressure and harmonic for body force) being the most stable, while the shifted pressure the most unstable. For the unstructured mesh, the shifted pressure formulation gets more deteriorated as can be shown in the pressure distribution figure, while highly deteriorated in the pressure magnitude distribution. The other methods have shown good agreement with analytic results for pressure and pressure gradient magnitude. As for the velocity magnitude distribution, high fluctuations are observed for all the methods, except for the method where the body force is reconstructed



(a) (b)

Figure 9.5: Schematics of the water-air phases at equilibrium test cases for meshes of type: (a) dense cartesian and (b) dense triangular. *Dimensions in m*

using linear interpolation for pressure and body forces. The combined traits method have shown less stable velocity at the interface than the linear-linear reconstruction method.

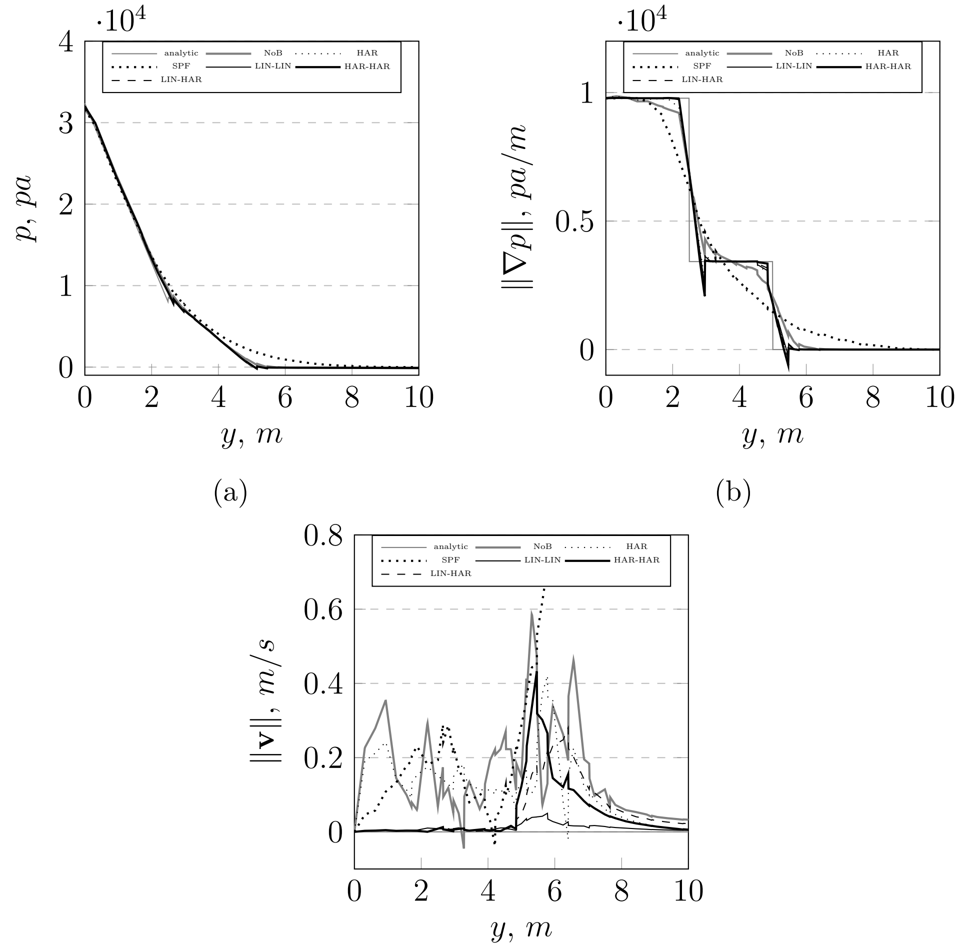


(a)

(b)

(c)

Figure 9.6: Results of the dense cartesian grid: (a) pressure distribution, (b) pressure gradient distribution, (c) velocity magnitude distribution. The abbreviations used in the figure are defined as follows. NoB corresponds to no force balancing, HAR corresponds to density-weighted harmonic interpolation of the pressure, SPF is the shifted (modified) pressure formulation, and LIN-LIN, HAR-HAR and LIN-HAR are the methods which involve reconstructing the body forces by either using a linear interpolation for body forces with a linear interpolation for the Rhie-Chow terms, harmonic for the first and the second, or linear for the first and harmonic for the second.



(a)

(b)

(c)

Figure 9.7: Results of the dense unstructured grid: (a) pressure distribution, (b) pressure gradient distribution, (c) velocity magnitude distribution. The abbreviations (NoB, HAR, etc) are defined in the previous figure.

It is clear from the results that the reconstruction of body forces using a linear interpolation with a harmonic interpolation in the Rhie-Chow has generated the most stable interface. And it is superior to the other methods. This is the method to be incorporated in the coupled VOF solver throughout.

9.2 Interface Compression

The ability of the solver to produce an accurate interface is a critical issue to assess. In this section the objective is to assess the capability of the solver to compress the interface using various approaches. Two test cases are simulated to test the interface accuracy, a rotational flow advecting a slotted circle, and another rotational flow advecting a circle of water with the vortex being centered away from the disk center; the latter is referred to as shear flow test [106].

9.2.1 Rotation of a Slotted Circle

This test case, originally defined by Zalesak [107], considers a slotted circle in a rotational flow. A schematic is shown in figure 9.8. The velocity components are defined as:

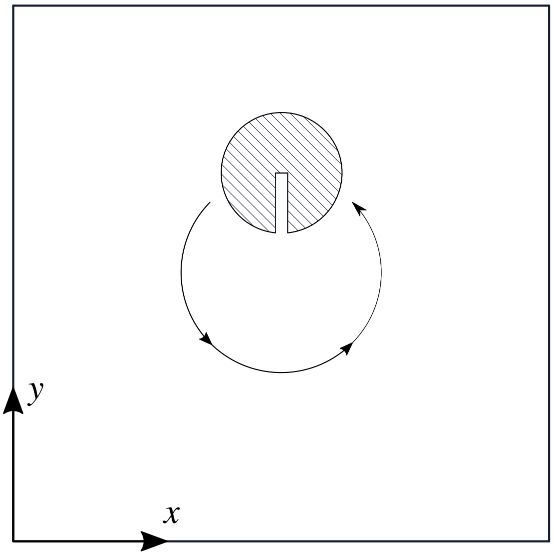
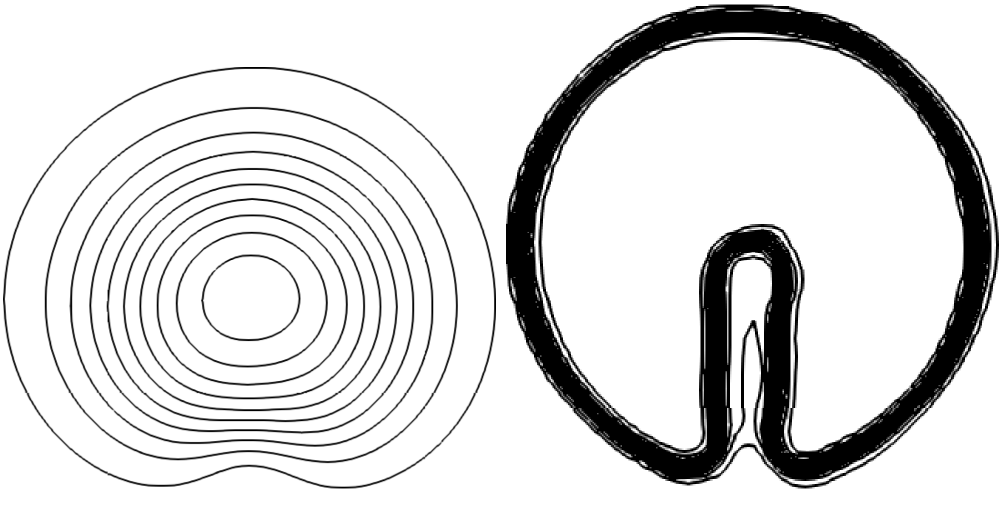


Figure 9.8: The rotation of a water-filled slotted circle schematic.

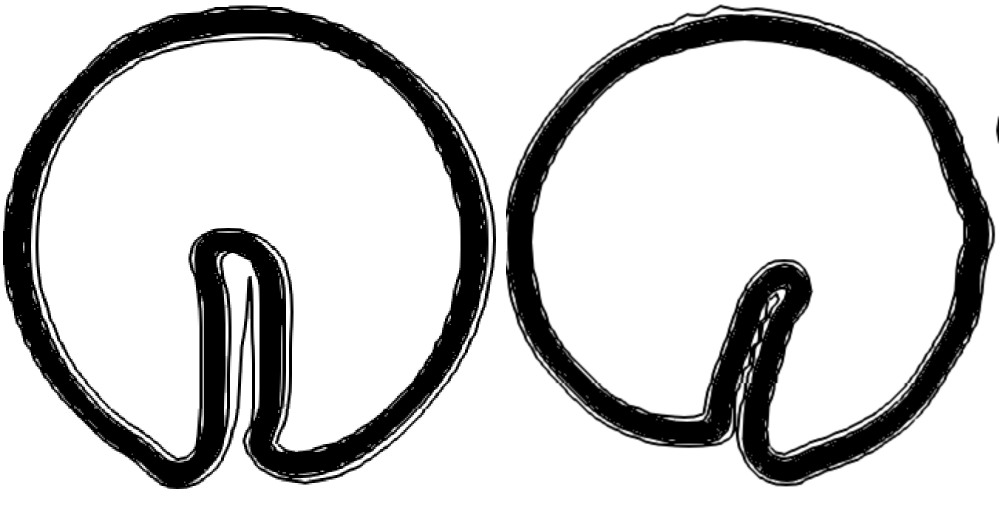
$$u = -\omega(y - y_0); \qquad v = \omega(x - x_0) \tag{9.5}$$

where ω is the angular speed, set to 0.5 rad/s, while x_0 and y_0 are the coordinates of the center of the slotted disk, which are 2 and 2.75. The computational domain is a 4 × 4 square divided into 200 × 200 uniform elements. The total time of the simulation is that required for the slotted circle to complete 1 revolution, which is $2\pi/\omega$. Two Courant numbers are set for this simulation, a low and a high one corresponding to two time steps 0.00298 s and 0.0298 s respectively as classified by [3]. Interface compression is attempted using the STACS advection scheme, the Barth and Jesperson with Downwind Boundedness Principles and also using the interfacial anti-diffusion technique with maximum compression ($\gamma = 1$).

Figures 9.12 and 9.13 present contour lines of an advected arbitrary scalar corresponding to a low Courant number and a high Courant number respectively. Contour lines are plotted in figure 9.12 for the low Courant number runs and in figure 9.13. It can be obviously seen that all three schemes have worked well for low Courant number except for the anti-diffusion methodology where it showed a sharp interface yet not an accurate one. For high Courant numbers, STACS and Barth and Jesperson schemes have produced less accurate shape of the slotted circle as compared to a low Courant number in addition to some wrinkles appearing in the circle itself, but in general, they have preserved to some acceptable extent the slotted circle. The anti-diffusion method have preserved the same performance as it did for a low Courant number. However, the shape of the slotted circle as produced by the anti-diffusion method is a bit deteriorated, up to the same level of deterioration that took place for a low Courant number. It can be concluded that the anti-diffusion method is the least sensitive to Courant number because it maintained a sharp interface, but it does not greatly preserve the physical shape. STACS as well as Barth and Jesperson schemes are more sensitive to Courant number, but they better preserve the shape of the interface.

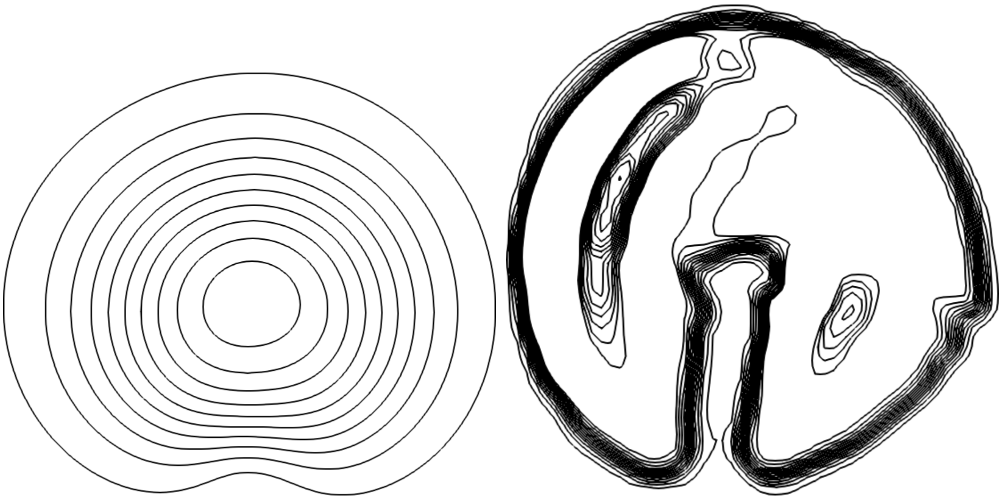


(a) Upwind (b) STACS

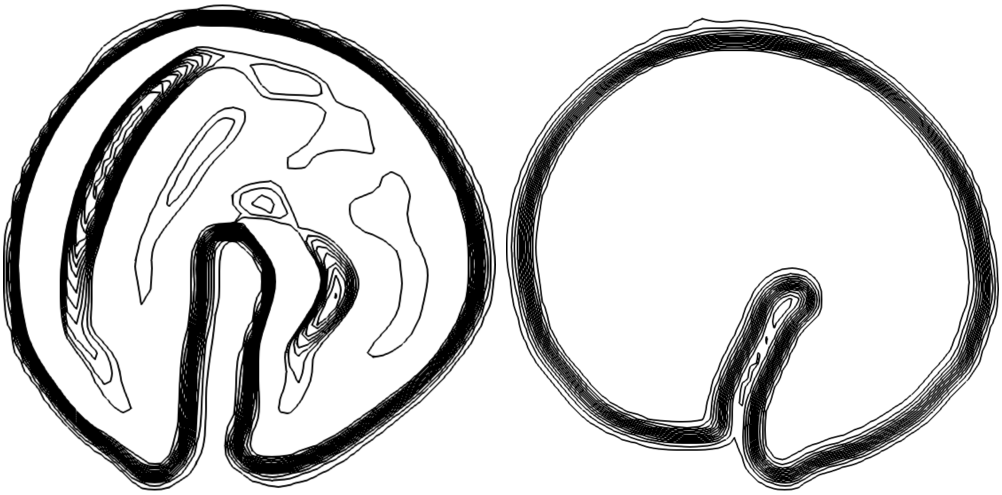


(c) Barth and Jesperson with Downwind Boundedness (d) Anti-diffusion with maximum compression

Figure 9.9: Contour lines at low Courant number for the slotted circle after 1 revolution. The contours are displayed for $0.05 < \alpha^k < 0.95$ with 13 contour lines



(a) Upwind (b) STACS



(c) Barth and Jesperson with Downwind Boundedness (d) Anti-diffusion with maximum compression

Figure 9.10: Contour lines at high Courant number for the slotted circle after 1 revolution. The contours are displayed for $0.05 < \alpha^k < 0.95$ with 13 contour lines

9.2.2 Single Vortex Flow

The single vortex problem [108, 109] is widely used as a benchmark. In consists of a rotating fluid body in a 1×1 m square. A vortex is defined by the stream function,

$$\psi = \frac{1}{\pi} \sin^2(\pi x) \cos^2(\pi y) \cos\left(\frac{\pi t}{T}\right) \tag{9.6}$$

and the velocity components are derived such as:

$$u = -\frac{\partial \psi}{\partial y}, \qquad v = \frac{\partial \psi}{\partial x} \tag{9.7}$$

These above velocity components are imposed in the domain and they are constant throughout time. On another hand, it is worth mentioning that this case is a pure advection problem but with strong shearing effects because the rotational velocity is not uniform in the radial direction, making it a different benchmark than the slotted circle problem.The mesh consists of 128 × 128 structured elements, and all the boundaries of the domain are symmetry planes. Initially, the domain consists of a circle of radius 0.15 m filled with water and centered at $(0.5, 0.75)$. Figure 9.11 shows the initial state of the water-filled circle. The results for low Courant number are shown in figure 9.12 while those for high Courant number are shown in figure 9.13. For both cases, low Courant and high Courant numbers, the results obviously show that STACS as well as Barth and Jesperson schemes have successfully captured the interface of the rotating body and preserved its shape and sharpness, unlike the anti-diffusion method which has given rise to wrinkles as well as heavy deterioration in the domain. This leads to a conclusion that the anti-diffusion, in its current formulation, is not able to handle a sharp and accurate interface for flows that exhibit shearing effects.

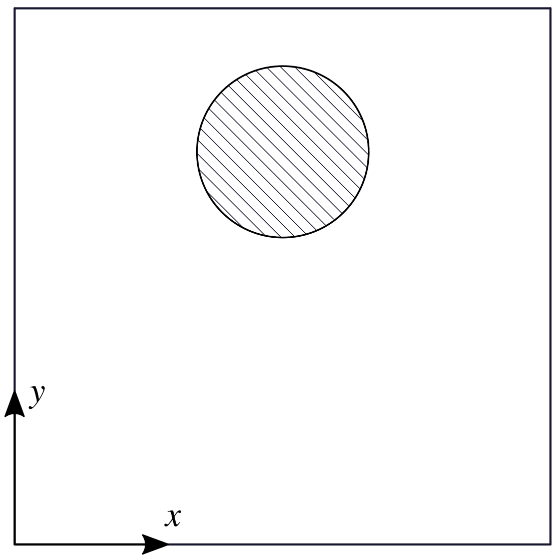
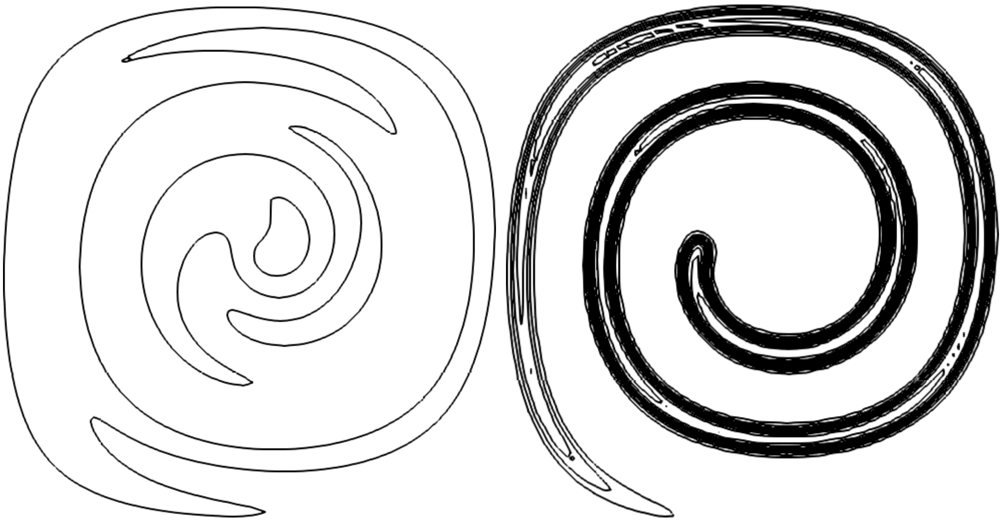
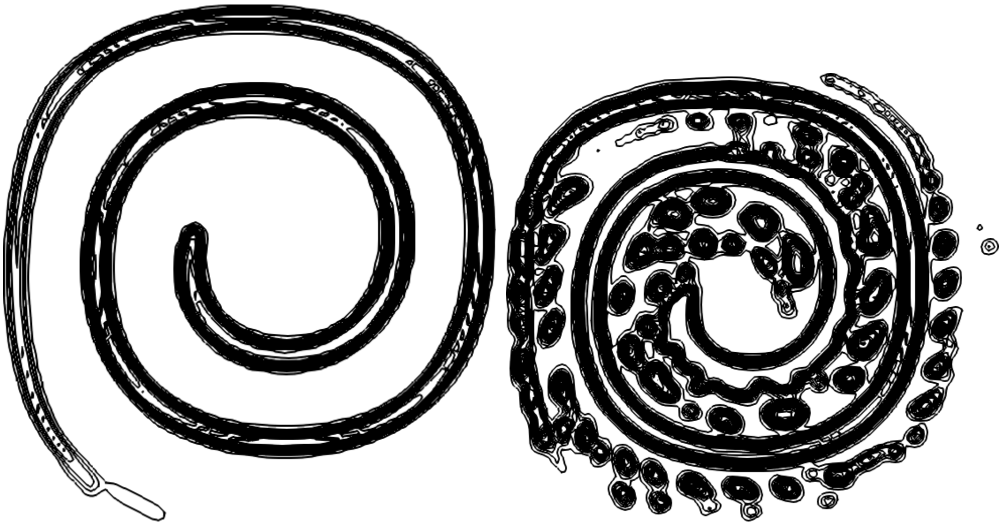


Figure 9.11: The shear flow test case at initial state

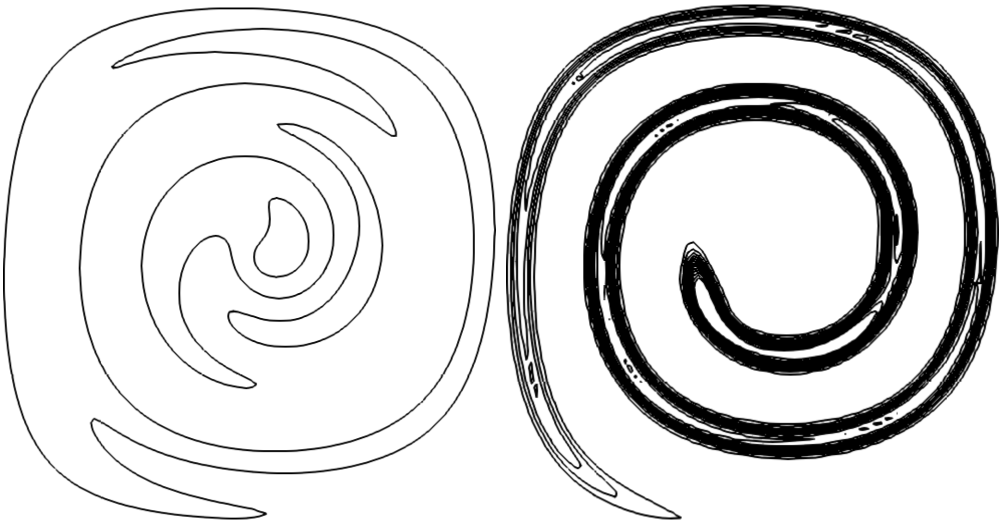


(a) Upwind (b) STACS

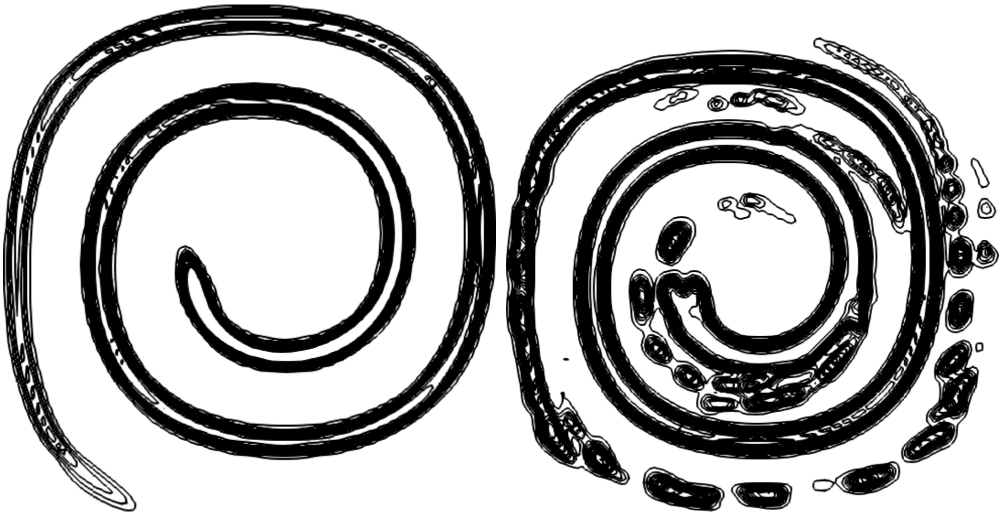


(c) Barth and Jesperson with Downwind Boundedness (d) Anti-diffusion with maximum compression

Figure 9.12: Contour lines at low Courant number for the shear flow after 3 *sec.* The contours are displayed for $0.05 < \alpha^k < 0.95$ with 13 contour lines



(a) Upwind (b) STACS



(c) Barth and Jesperson with Downwind Boundedness (d) Anti-diffusion with maximum compression

Figure 9.13: Contour lines at high Courant number for the shear flow after 3 *sec*. The contours are displayed for $0.05 < \alpha^k < 0.95$ with 13 contour lines

9.3 Geometric Conservation

In this technique assessment test, the ability of the coupled VOF solver to satisfy the geometric conservation constraint as discussed previously is monitored. The coupled VOF solver is equipped with the volume fraction prediction alternatives reported previously (explicit geometric conservation constraint and mutual influence of volume fractions with explicit/implicit normalization). The reported results here consider the implicit normalization method to fulfill the geometric conservation constraint. The selected test case is a flush tank problem (figure 9.14a) where a mass flow rate at the inlet is fixed to 0.2 kg/s for 1 seconds. The case is an unsteady 2D problem with a quadrilateral mesh of 16928 elements, shown in figure 9.14b. Scaled residuals of the method through the simulation are plotted in figure 9.15, signifying an acceptable level of convergence for the simulation using the coupled VOF solver. To monitor a representative geometric conservation quantity over the simulation time, a parameter $\mathcal{S}$ is defined as

$$\mathcal{S} = \sum_{k=1}^{n} \alpha^k \tag{9.8}$$

Maximum and minimum values of $\mathcal{S}$ are of interest. Figure 9.16 shows $\mathcal{S}$ over the internal iterations of the first time step in the simulation, and it clearly depicts that the implicitly normalized procedure preserves geometric conservation throughout. Figure 9.17 shows contour samples of the first internal iterations within the first time step of simulation. The CONSTRAINED column refers to the cases where the implicit normalization is incorporated in the solver, while the UNCONSTRAINED column refers to the case where no constraint is applied.

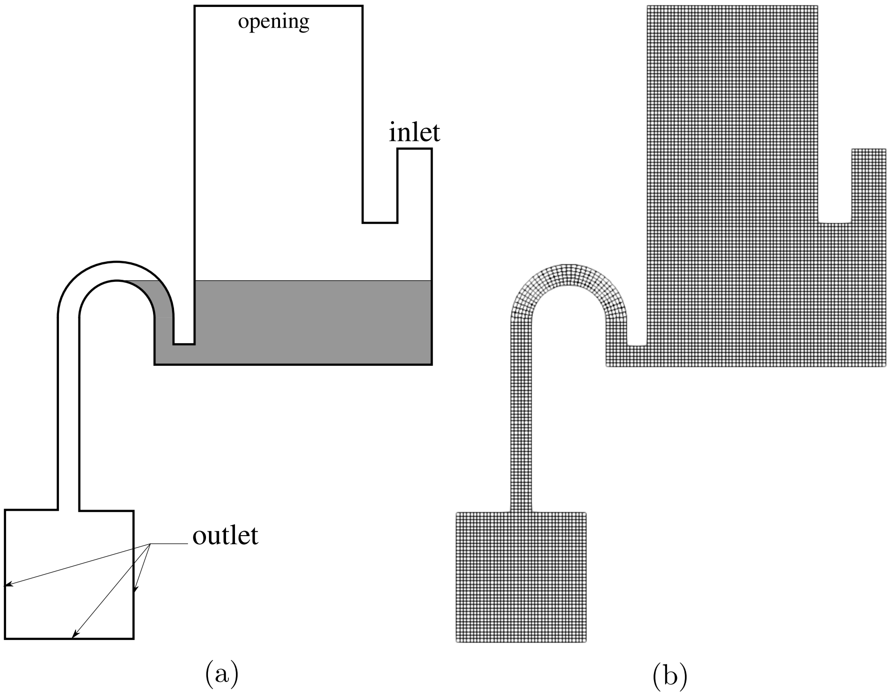


Figure 9.14: The flush tank case (a) schematic and (b) mesh

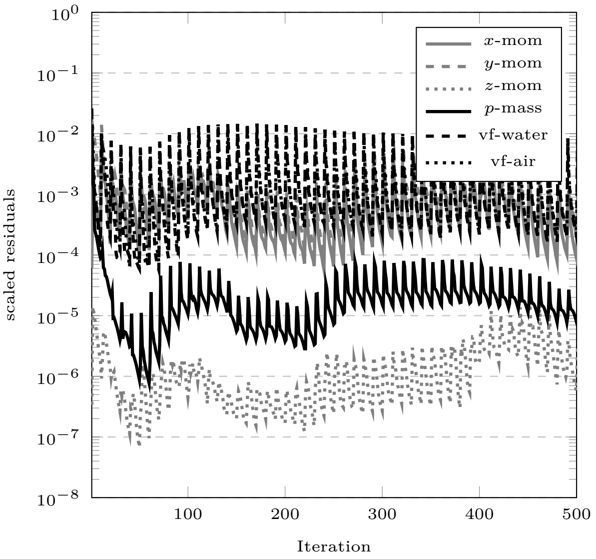


Figure 9.15: RMS Scaled residuals versus iterations through simulation.

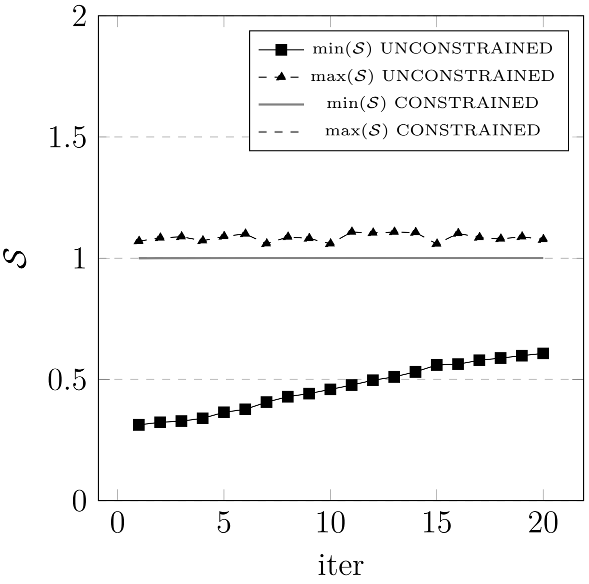


Figure 9.16: Geometric conservation throughout the internal iterations within the first time step of simulation time.

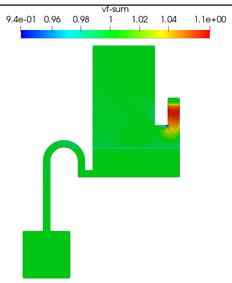
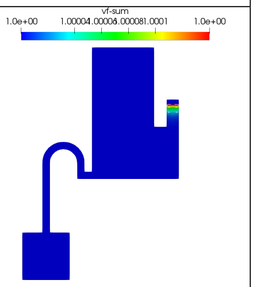
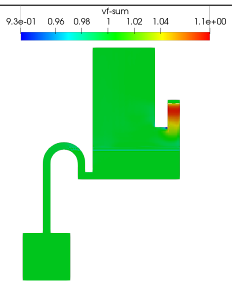
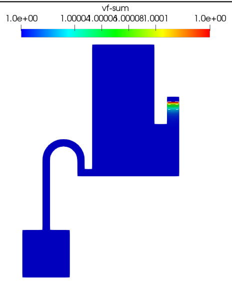
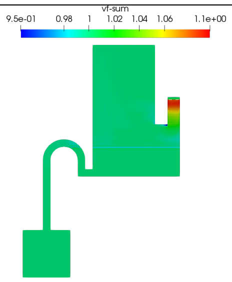
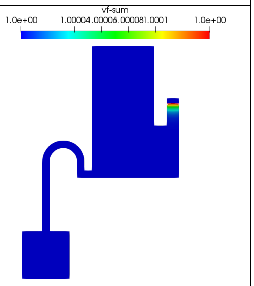
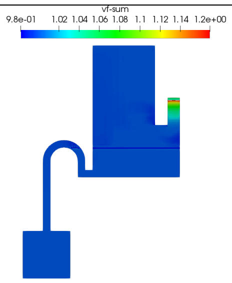
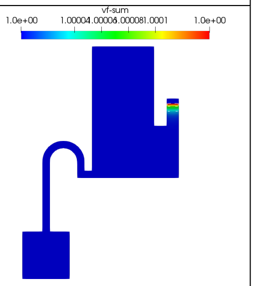
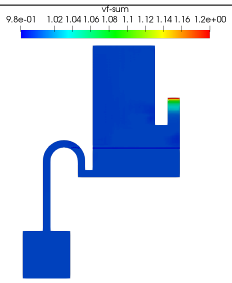
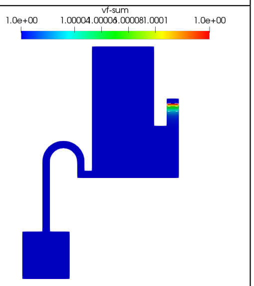
ITERATION	UNCONSTRAINED	CONSTRAINED
1		
2		
3		
4		
5		

Figure 9.17: Geometric conservation parameter $\mathcal{S}$ contours over the flush tank surface for the first 5 iterations

Chapter 10

Test Cases

The current section includes engineering simulation tests to validate the performance of the coupled VOF solver. In the set of test cases below, two transient cases and two steady-state engineering cases are considered for evaluation. The performance of the coupled VOF solver is evaluated with two criteria; the convergence rate as compared to a fully-coupled VOF solver from a commercial software (ANSYS Fluent), and the accuracy of the solution as compared to that of the commercial software as well as available experimental data. It has to be clarified that for some cases, the convergence rate might not reflect alone the exact potential of the solver due to certain numerical features or physical aspects. For example, the linear solver in the commercial solver might employ different scaling and normalizing standards, thus, its convergence could be misleading. This is very prominent in transient cases as will be shown. Another example is related to open channel flows over an obstacle. The free surface in an open channel, depending on the flow conditions and geometry, might exhibit very unstable surface waves, and it takes a long time to damp at some locations in the domain. With this example, it is clear that the convergence rate might be influenced because some interfacial zones in the domain are still unconverged or unresolved, while this behavior may not reflect for example in the level of drag over the bump. By all means, the success of the solver can then be identified by selecting proper monitors, such as the drag.

The simulations are run with reasonably selected numerical methods and controls (rather than the volume fraction coupling), addressed within the description of each of them. These methods and controls are to be matched as much as possible between the competing solvers. For instance, the use of the same physical time step or pseudo time step is crucial. Also, the use of the same advection schemes is also important because it is known that higher-order advection schemes may deteriorate the stability of the solver unless more relaxation is introduced. Under-relaxation, explicit relaxation, Courant number limitation, number of iterations per time step, etc are all numerical controls to be matched. It is also very important to mention that the use of the same boundary conditions is also a key for a consistent and fair evaluation.

10.1 Forced Sloshing in Water Tank

This case considers a forced sloshing problem in a water tank, originally reported by [110]. These cases are usually investigated for purposes related to tuned liquid dampers when exposed to excitation; this is shown in a schematic in figure 10.1. The case is presented here to validate the credibility of the solver. The tank, shown in figure 10.2, undergoes horizontal oscillation such that the acceleration is $a = A\omega^2 \sin\omega t \; m/s^2$, where $A = 0.02\; m$ and $\omega = 2.877\; rad/s$. The computational domain consists of 43770 structured elements with hanging nodes. A transient run is made with a fixed time step 0.01 s for 12 s. Laminar conditions are assumed in this case and compressive advection schemes are set for volume fraction equations to accurately capture the interface. The run is made using the current coupled VOF solver, and is also made using a coupled VOF solver from a commercial benchmark; the use of the same numerical methods and control settings in both solvers is a necessity to render fair evaluation. A sample of the scaled residuals is provided by the convergence history plot in figure 10.3. The scaled residuals for both solvers appear to be quite close and comparable. On the other hand, the height of the water surface is monitored in time at the left wall of the tank. This is shown in figure 10.4, along with experimental data and the solution from a commercial solver. The water levels as predicted by the solvers are very close to each other and consistent at the same time with experimental data.

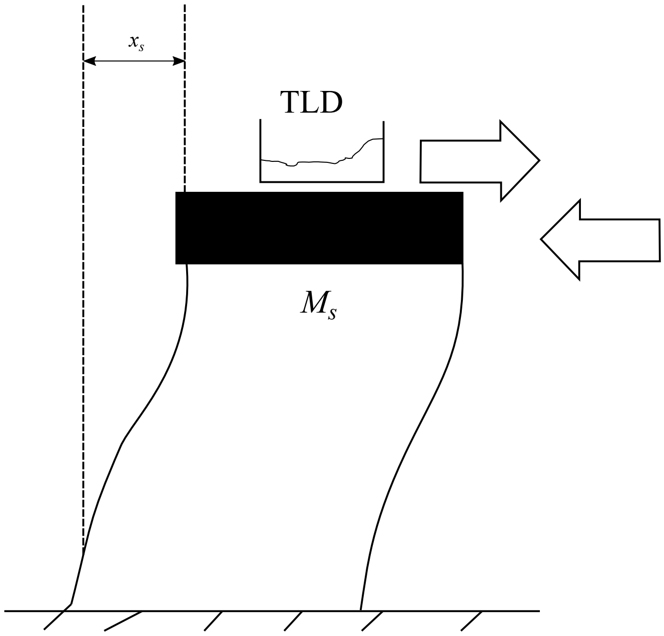
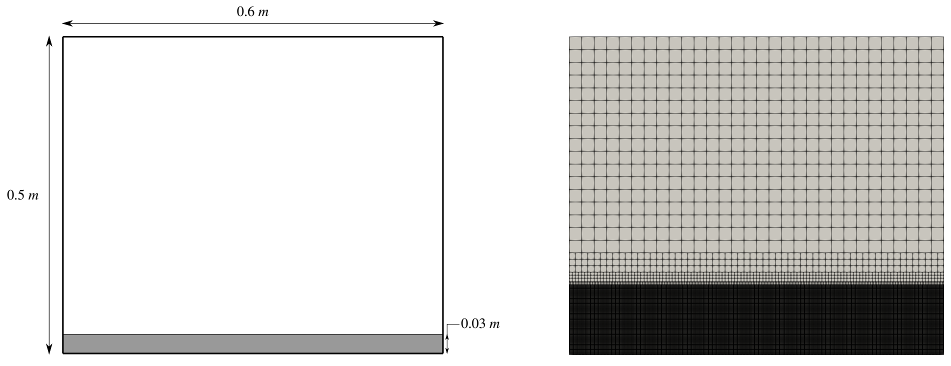
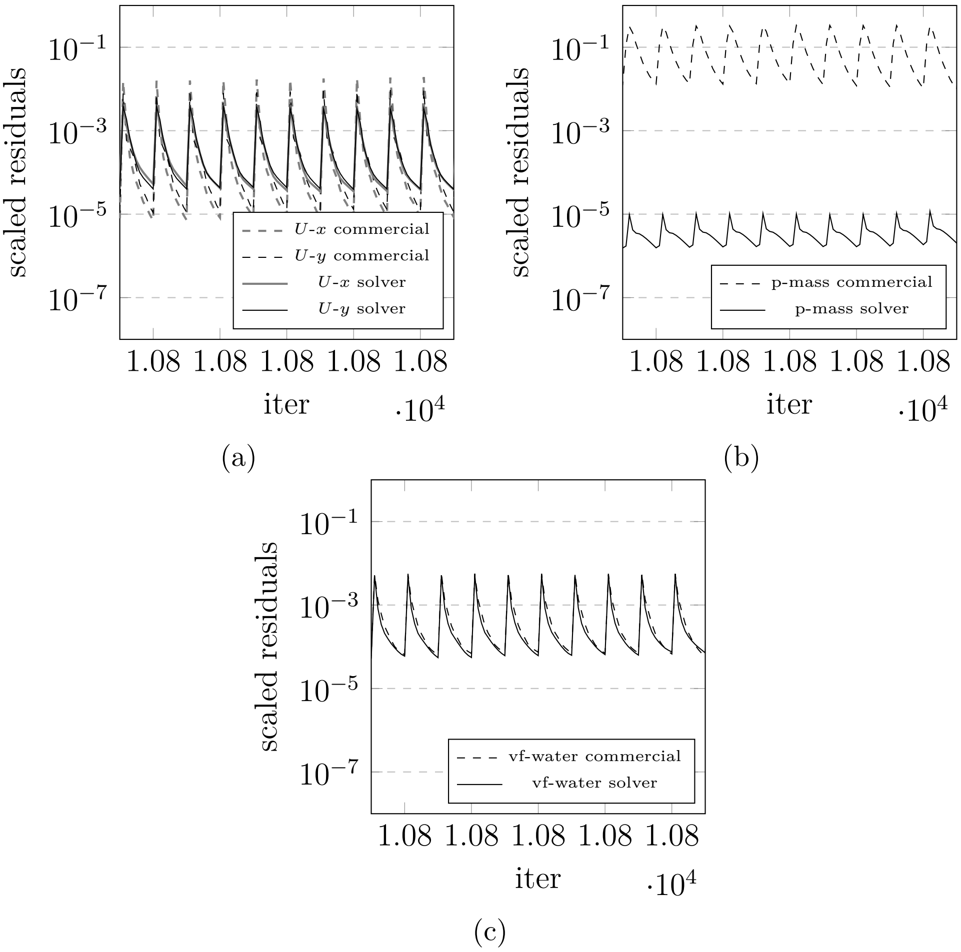


Figure 10.1: Schematic of tuned liquid damper attached to single-degree-of-freedom structural system. x_s refers to the displacement of the structural system and M_s refers to the mass of the structural system



(a) (b)

Figure 10.2: The forced sloshing case (a) geometry and (b) mesh



(c)

Figure 10.3: Sample of scaled residuals for the forced sloshing case: (a) momentum equation, (b) mass equations and (c) water volume fraction equation

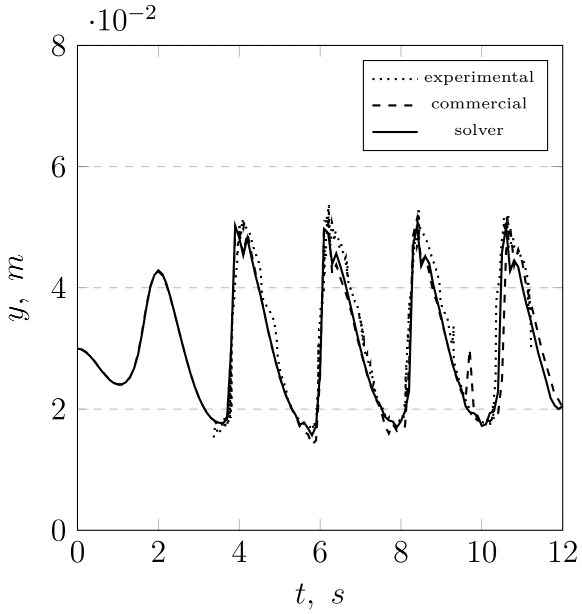
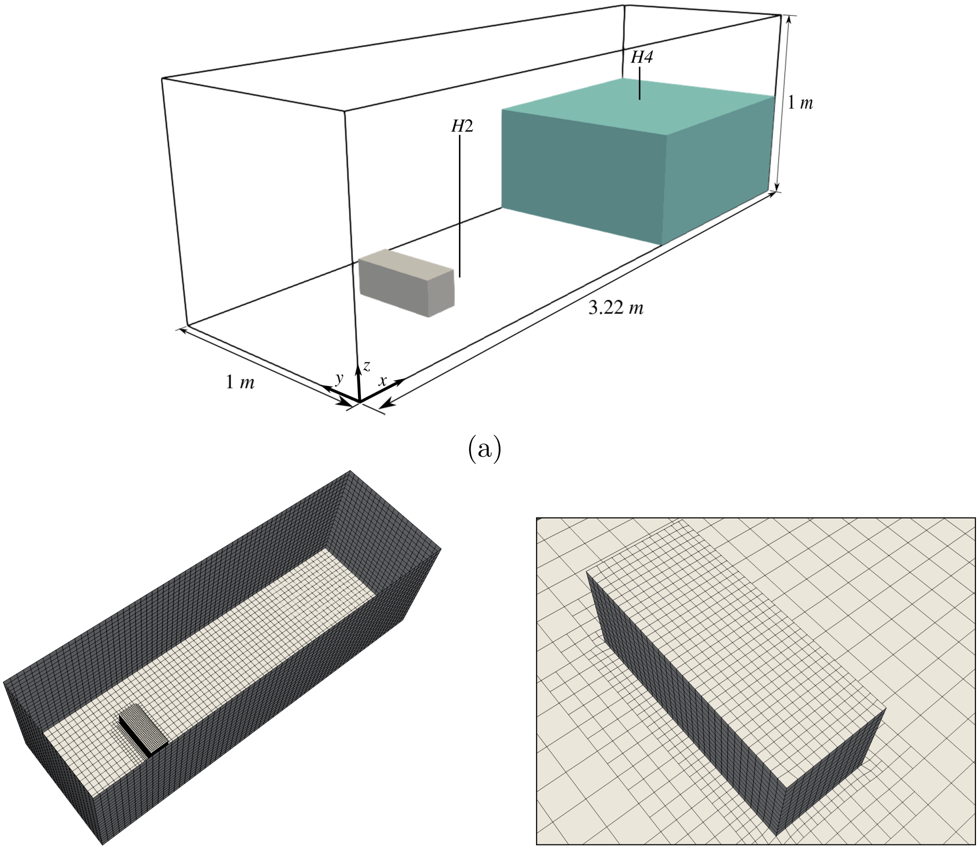


Figure 10.4: Water level at left edge throughout simulation

10.2 3D Dam Break Problem

This case is a very widely simulated problem due to its easy setup [111, 112, 113]; it is considered a simple model of green water flow on the deck of a ship. The purpose of this test case is to assess the accuracy of the developed solver for problems involving wave impacts. The model includes a large tank of dimensions $3.22 \times 1 \times 1\ m$ with an open roof. Figure 10.5a shows a schematic of the case. A water column of height 0.55 m and length 2.22 m, initially at rest, collapses under the action of gravity and flows into the empty part of the tank, while hitting an obstacle as shown in the figure. The mesh, shown in figure 10.5b and 10.5c, consists of 31280 hex-dominant elements. The time step is set to $dt = 0.01\ s$ and $k - \epsilon$ turbulence model is used. The Barth and Jesperson high resolution scheme is applied with downwind behavior and is set as the advection scheme for the volume fraction equations. The time step is fixed to 0.01 s. The scaled residuals of the simulation with the developed solver along those generated by a simulation with the aforementioned commercial solver are presented in figure 10.6. The residuals in the commercial solver show lower drop rates per time step, but however, this is not the case for the continuity equation. By all means, a better insight into the convergence of the simulations can be carried by monitoring the water level at two locations in the domain denoted by H2 and H4, and indicated in figure 10.5a. The monitored water levels using the developed solver versus those generated by the commercial solver are depicted in figure 10.7. The experimental data are retrieved from [113]. The water level progress as predicted by the solver at both locations is very consistent with that predicted by Fluent and established experimentally. Also, the free surface colored with velocity magnitude values is plotted in figure 10.8 in a set of consecutive figures alongside with corresponding results by the commercial solver; it can be noticed obviously that the velocity variations over the free surface as predicted by both solvers are close.

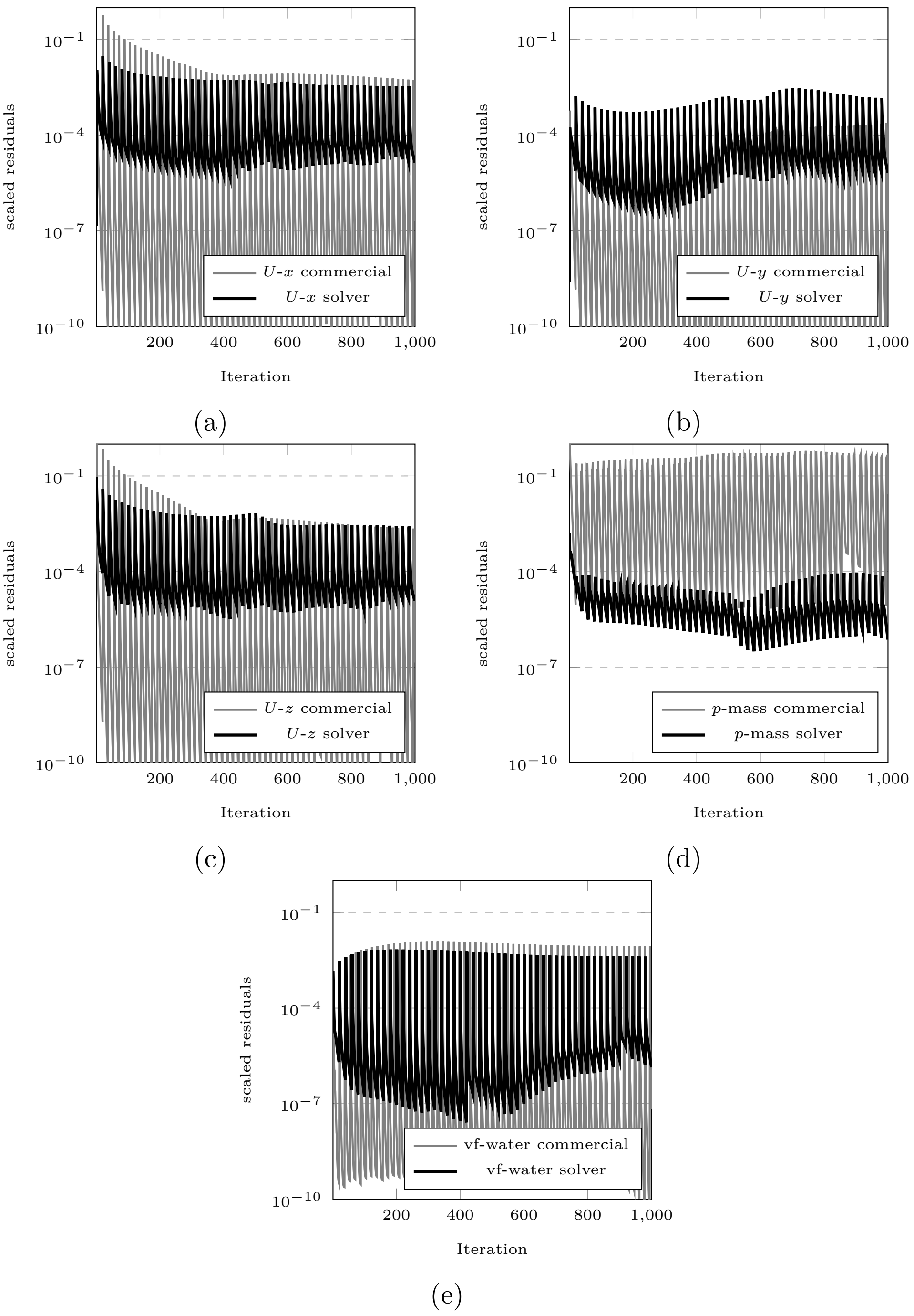


(a)

(b)

(c)

Figure 10.5: The 3D dam break (a) schematic and (b) mesh-overview (c) mesh at obstacle



(a)

(b)

(c)

(d)

(e)

Figure 10.6: Scaled residuals versus accumulated iterations over 1 *sec* of simulation time for the developed solver and the commercial solver

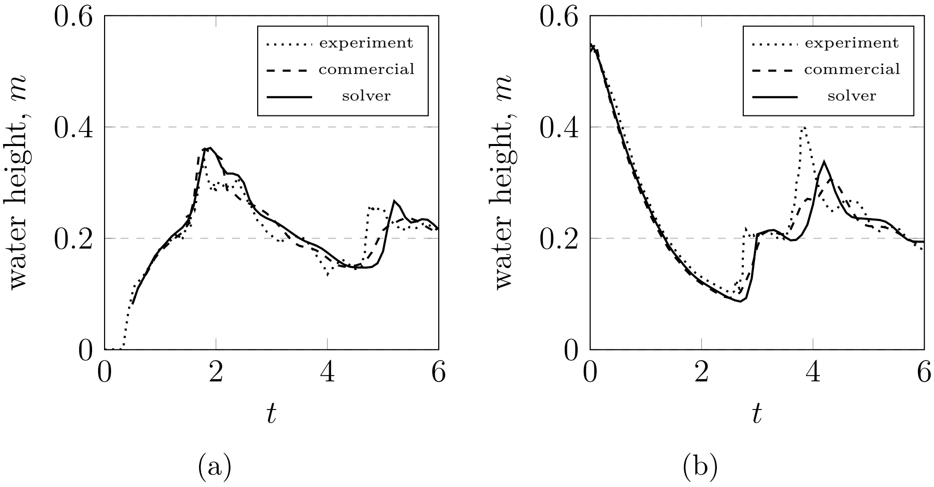


Figure 10.7: Vertical water heights as predicted by the developed solver and the commercial solver at the (a) tank H2 and the (b) reservoir H4

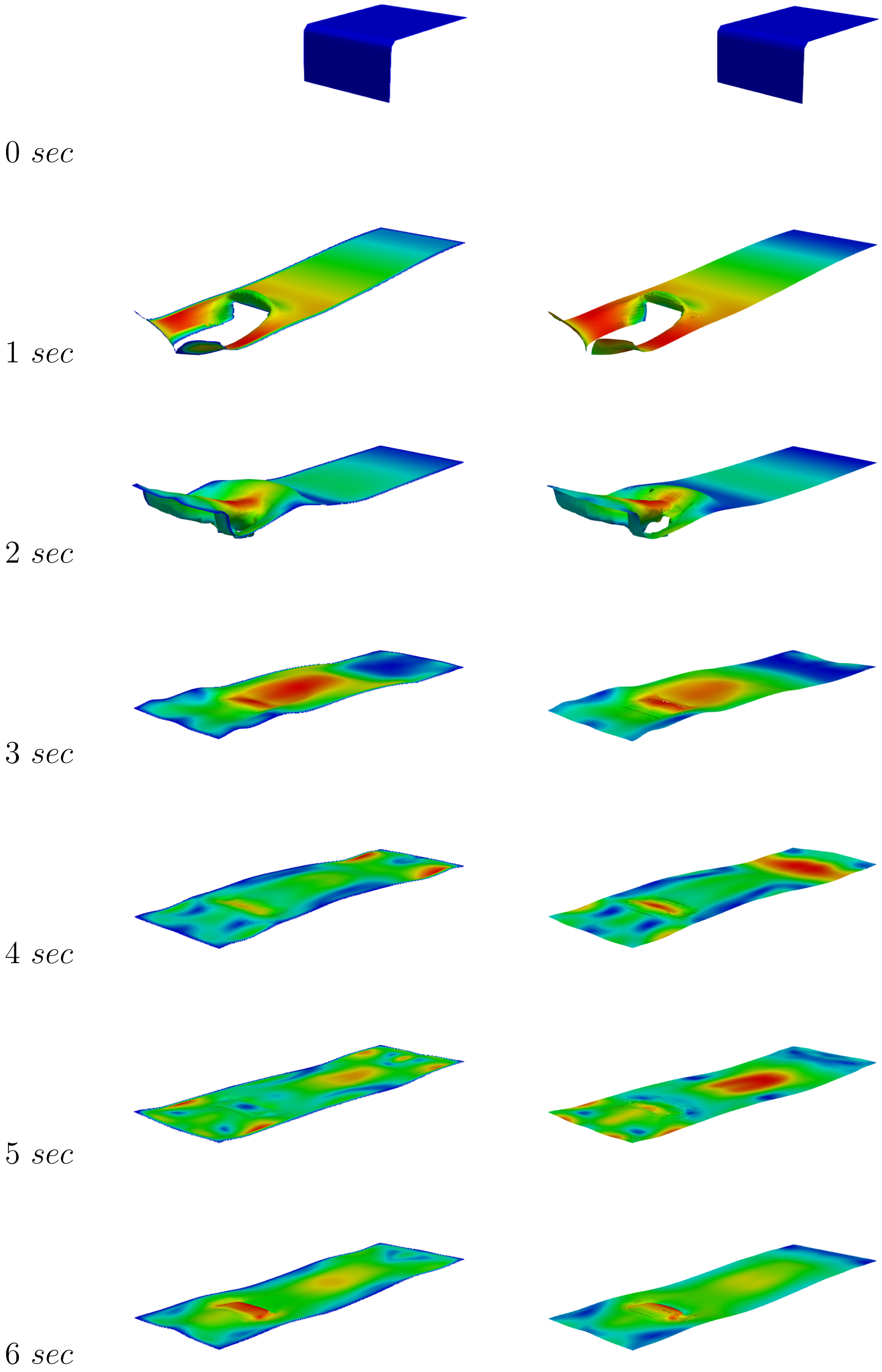


Figure 10.8: Time progress of the free surface throughout simulation time colored with velocity contours, as predicted by the developed solver (left column) and the commercial solver (right column)

10.3 Water Flow past a Vertical Surface-Piercing Circular Cylinder

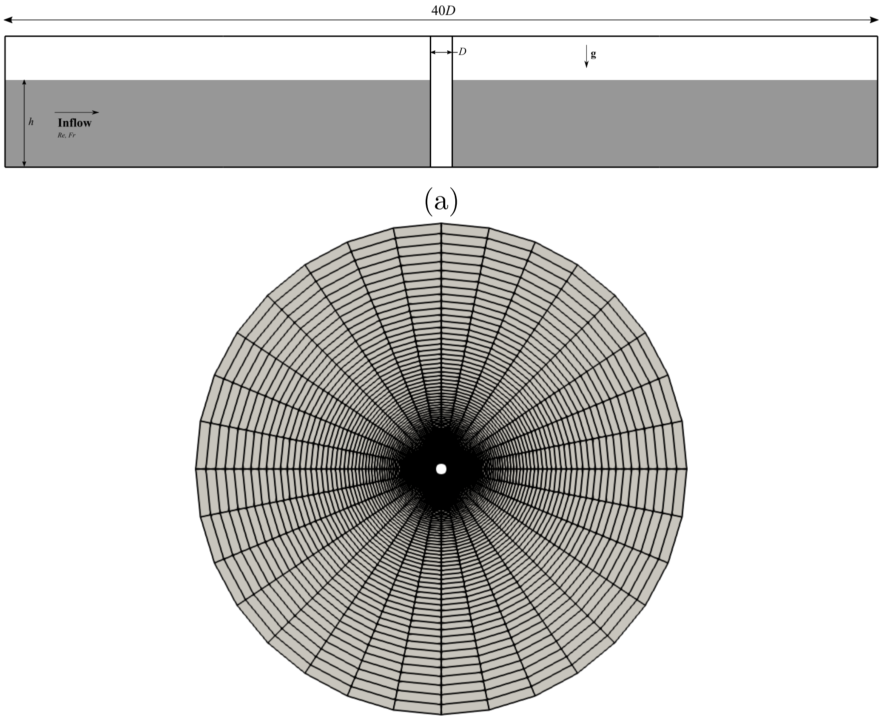
The current case considers a turbulent free surface flow past a vertical surface-piercing cylinder. This case has been widely investigated due to its importance in many hydrodynamic engineering applications [114, 115, 116, 117, 118, 119]. A schematic is shown in figure 10.9a. The flow at various Reynolds numbers (Re) and Froude numbers (Fr) is studied. Re and Fr are defined as

$$Re = \frac{\rho U_\infty D}{\mu} \qquad Fr = \frac{U_\infty}{\sqrt{gD}} \tag{10.1}$$

where U_∞ is the free-stream velocity of the water stream at the inflow. D is the cylinder diameter and it is equal to 0.8 m. The drag coefficient C_D over the cylinder is monitored for a set of flow conditions. C_D is defined as

$$C_D = \frac{F}{\frac{1}{2}\rho U_\infty^2 hD} \tag{10.2}$$

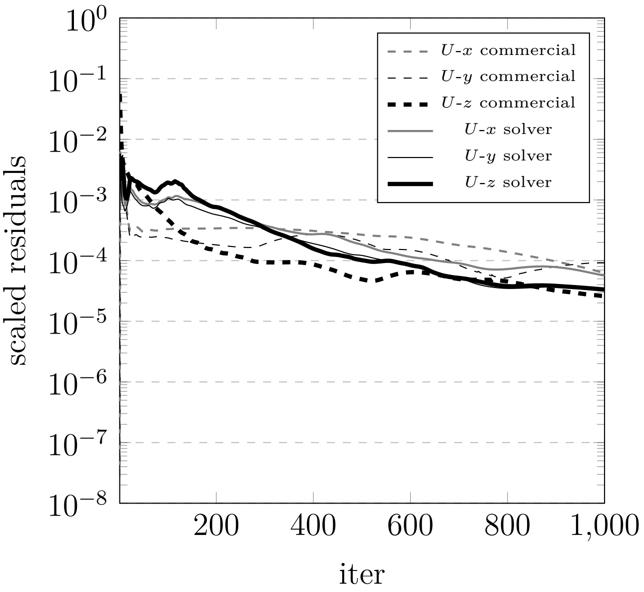
F is the axial force the flow exerts on the cylinder and h is the height of the flow stream at inflow. In this case, h is fixed to $4D$. The domain and mesh are as shown in figures 10.9a and 10.9b. The domain diameter is $40D$ and its height is $6D$, and the mesh consists of 102400 structured elements. The simulations were done using the available coupled VOF solver and also generated using commercial code. Scaled residuals for the current solver and the commercial code are plotted in figure 10.10 for a selected Fr and they show close performance. The corresponding drag history as predicted by both solvers is plotted in figure 10.11. The drag histories for both solvers are said to have close trends towards the right drag value over the cylinder. For a set of Fr, C_D data are plotted in figure 10.12. Experimental results from [119] are retrieved for validation purposes. The drag characteristics as a function of Fr as generated by both solvers are very similar, and are also consistent with experimental data.



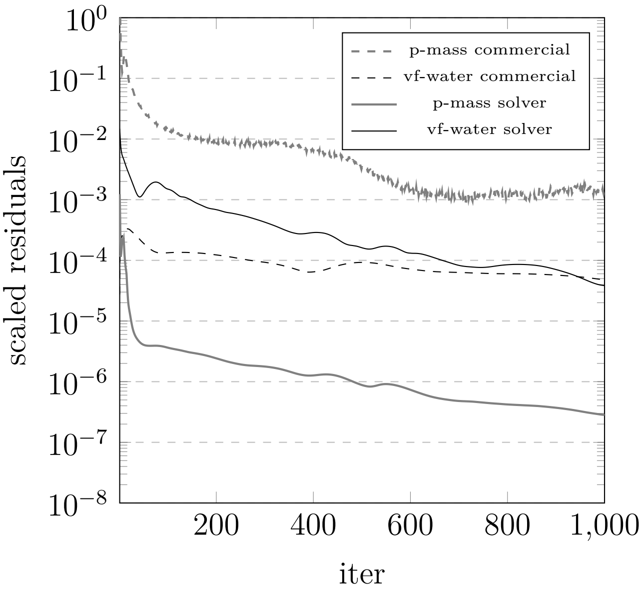
(a)

(b)

Figure 10.9: The flow past cylinder (a) geometry and (b) top view of mesh



(a)



(b)

Figure 10.10: Sample of scaled residuals for the flow past cylinder case: (a) momentum equation and (b) p-mass equation and vf-water equation

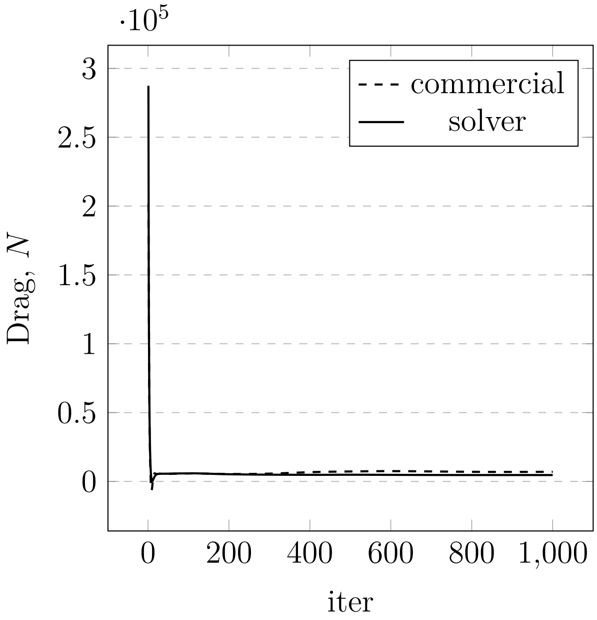


Figure 10.11: Drag history over cylinder throughout the simulation

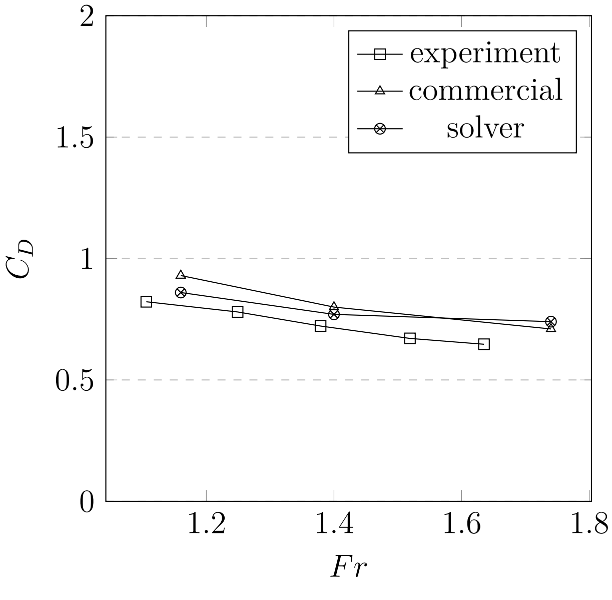
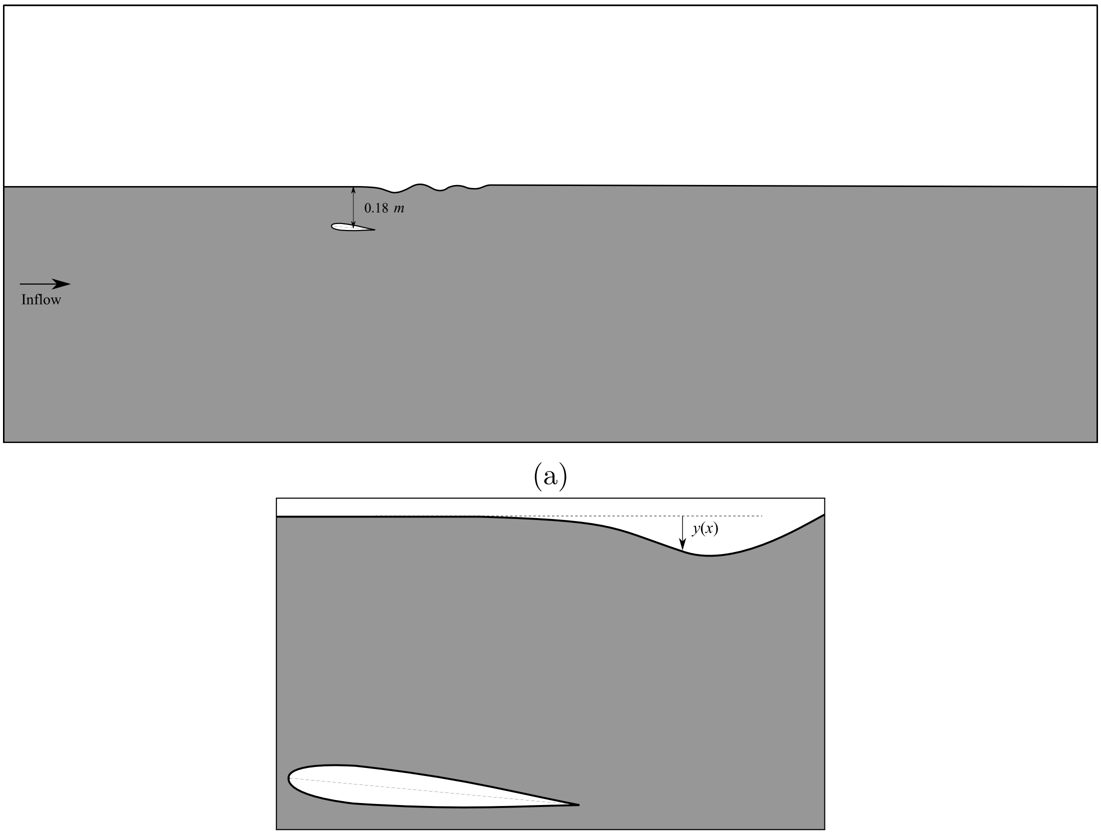


Figure 10.12

10.4 Flow over a Submerged Hydrofoil

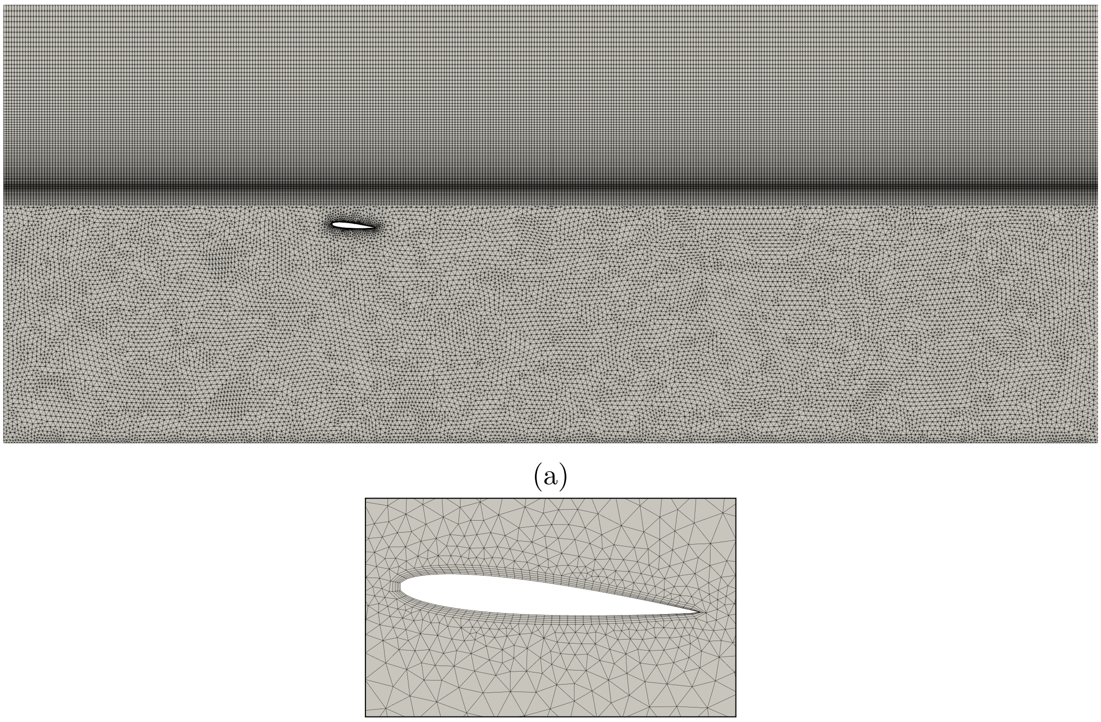
The wave generation as a result of a steadily moving body beneath a free surface has been an active area of research [120, 121, 122, 123] in marine hydrodynamics. The current test case considers a submerged hydrofoil, a very popular experiment established by Duncan [124] to measure the wave profiles and estimate the drag over the hydrofoil. A makeup of the experiment is presented here. The schematic of the test case is shown in figure 10.13. A NACA0012 foil is incorporated, with a chord length of 20 cm. The domain size is 25 times the chord length of the foil in longitude, while its height is set to 2 m to keep a sufficient distance above the free surface. The foil is placed at a height of 1 m from the domain's bottom, while the free surface is at a height of 0.18 m from the hydrofoil (measured from the midpoint of the chord). The inflow has a speed of 80 cm/s. The corresponding mesh of the domain consists of 70719 mixed elements (Figure 10.14). The SST $k-\omega$ model is activated for modeling turbulence. The case is run using coupled VOF solvers from the developed framework and from a commercial package with pseudo transience of 0.01 s to stabilize the simulation, and compressive schemes are utilized to capture the interface accurately. Drag monitors over the hydrofoil are plotted in figure 10.15. Both solvers have succeeded in approaching a steady-state drag over a short time (~ 0.46 N). On another hand, the wave profiles of the free surface are plotted in figure 10.16. The wave profiles are quite consistent with the experimental data provided by [124].



(a)

(b)

Figure 10.13: The flow over submerged hydrofoil case (a) schematic and (b) wave schematic



(a)

(b)

Figure 10.14: The flow over submerged hydrofoil (a) overall mesh and (b) mesh at the hydrofoil vicinity

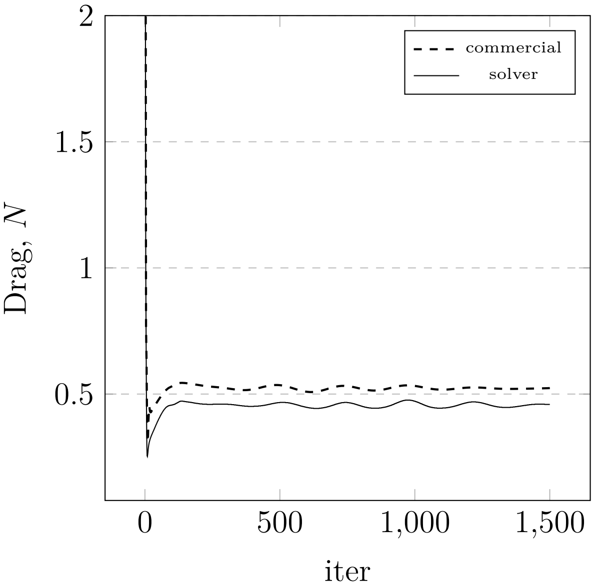


Figure 10.15: Drag histories over the hydrofoil

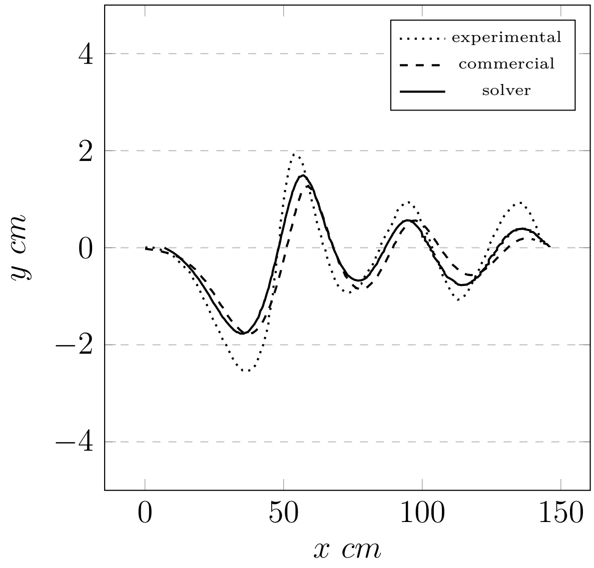


Figure 10.16: Wave profile as predicted by commercial solver and developed solver

Chapter 11

Concluding Remarks

In this work, a fully-coupled VOF solver is developed for an efficient solution of free surface flows. A number of critical and important milestones to reach a robust and credible VOF solver has been addressed, namely, a force-balanced fully-coupled VOF solver and selected compressive schemes for capturing accurately the interface. Implicit geometric conservation approaches are also addressed and incorporated in the solver. In addition, the solver features a set of appropriate turbulence models, on the top of which is the SST $k - \omega$ turbulence model. The fine components of the solver have been sufficiently addressed within this thesis, with emphasis going for the discretization procedure of the fully-coupled VOF system which mainly involves a Newton-Raphson linearization technique to make an efficient coupling of volume fraction equations with velocity and pressure fields possible. The coupled VOF solver has been applied for a set of test cases with the intention of assessing the validity of the techniques featured in the solver such as the force-balancing infrastructure, the interface capturing, and the geometric conservation. The technique assessment cases have been followed by a set of real-life engineering simulations of transient and steady-state type, and have been put into comparison with the results of the popular commercial software Ansys Fluent in addition to other retrieved experimental data. To make the comparison fair and consistent with the commerical solver, the simulation methods and controls in both solvers (developed solver and commercial solver) have been almost matched. The solver in Ansys Fluent is the fully-coupled VOF solver obviously. Ansys Fluent has been selected among other commercial packages because it is the closest in infrastructure to the current solver featured in this thesis, in the sense that it employs the finite volume method for discretization, and it includes a wide variety of schemes, methods and controls such that a fair and consistent comparison can be made possible. Last and not least, the coupled VOF solver has showed successful behavior for both transient and steady-state simulations. However, it is very important to mention at this point that the research area of coupled VOF is still in a rising stage. This is to point that the performance of a segregated VOF solver can be tuned by some restrictive pseudo-transience or local time stepping methods, relaxation techniques, or other efficient acceleration procedures in order to produce the same convergence of a coupled VOF solver. This is not a paradox. This is a gear to investigate more into the embedded potentials of the fully-coupled VOF solver, and scrutinize the manipulated techniques such as the linearization methods, interpolation methods, and other mathematical/numerical techniques. It is also a gear to find the range of applicability of the coupled VOF solver where it can behave superior to a segregated one. Finally, this PhD thesis has sufficiently addressed the fully-coupled VOF solver and uncovered a potential coupling strategy to the VOF equations and have proved to be successful and by all means a competitive solver to a widely used commerical software.

Appendix A

Nomenclature

A.1 Fields and Parameters

$\mathbf{v}$	Velocity
p	Pressure
p'	Shifted pressure
p_{ref}	Reference pressure
p_{rel}	Relative pressure
T	Temperature
H	Specific total enthalpy
α	Volume fraction
ρ	Density
ρ_0	Reference density
μ	Viscosity
λ	Thermal conductivity
C_p	Specific heat
$\mathbf{g}$	Gravitational acceleration
g	Magnitude of gravitational acceleration ($\equiv \|\mathbf{g}\|$)
x, y, z	Rectangular coordinates
t	Time
$\mathbf{R}$	Position vector
$\mathbf{B}$	Body force
τ	Stress
$\hat{\mathbf{n}}$	Interface unit normal
$\mathbf{S}$	Face normal vector
$\mathbf{n}$	Face unit normal vector
g_f	Face weight
κ	Interface curvature
V	Element volume
n	Number of phases

N	Number of Elements
Re	Reynolds number
Fr	Froude number
C_D	Drag coefficient

A.2 Superscripts

k	Phase index
$*$	Latest available value
$\circ$	Previous time step value
(n)	Previous iteration value
x, y, z	$x,\ y$ or z component of a vector

A.3 Subscripts

C	Element
F	Neighbour element
f	Face
b	Boundary face
n	Node
∞	Free-stream

Appendix B

Turbulence Modeling

While the solution of the governing equations for laminar cases does not raise particular difficulties, the simulation of turbulent flows presents significant problems. The chaotic motion causes a mixing of the various layers, and due to the increased momentum and energy exchange between the molecules and solid walls, turbulent flows lead to higher skin friction at the same conditions as compared to laminar flows. Direct simulations of the fluid flow (DNS) is only possible for simple flow problems due to the high computational cost. Therefore other modeling methods are used to account for the effects of turbulence. The two main groups are LES (Large Eddy Simulation) and RANS (Reynolds Averaged Navier Stokes) turbulence models. While the first named group computes large eddies directly, the second group uses models for all scales of turbulence [125].

In the following, the basics of the RANS approach are described and emphasis is given after that to the $k-\omega$ SST turbulence model as it is superior to most of the Boussinesq-based models.

B.1 RANS Approach

Reynolds introduced in 1895 an approach for the approximate treatment of turbulent flows. His method is based on decomposition (Reynolds decomposition) of the flow variables into a mean and a fluctuating part. The flow variables then can be written as:

$$\mathbf{v} = \overline{\mathbf{v}} + \mathbf{v}^{'} \quad \text{and} \quad p = \overline{p} + p^{'} \tag{B.1}$$

where the mean values are denoted with an over bar and the turbulent fluctuations by a prime. Mean values thereby are obtained by an averaging procedure over time or space. By assuming stationary turbulence, a time averaging $\left(\overline{\mathbf{v}} = \lim_{T\to\infty} \frac{1}{T}\int_{t}^{t+T} \mathbf{v}dt\right)$ is chosen. If it is applied to the Navier-Stokes equations, the following relations for mass and momentum conservation are obtained (assuming incompressible):

$$\nabla\overline{\mathbf{v}} = 0 \tag{B.2}$$

$$\frac{\partial\rho\overline{\mathbf{v}}}{\partial t} + \nabla(\rho\overline{\mathbf{v}}\overline{\mathbf{v}}) = \nabla(\overline{\tau}_{ij} - \tau_{ij}^{R}) \tag{B.3}$$

These equations are known as the Reynolds Averaged Navier-Stokes equations (RANS). Due to the Reynolds averaging an additional term called Reynolds-stress tensor τ_{ij}^{R} appears, which represents the transfer of the momentum due to turbulent fluctuations, while $\overline{\tau}_{ij}$ represents the laminar viscous stress tensor. It can be denoted with averaged strain rate tensor $\overline{S}_{ij}$ and the averaged velocity components as

$$\overline{\tau}_{ij} = 2\mu\overline{S}_{ij} = \mu\left(\nabla\overline{\mathbf{v}} + \nabla\overline{\mathbf{v}}^{T}\right) \tag{B.4}$$

The Reynolds-stress tensor is built with the velocity fluctuation parts

$$\tau_{ij}^{R} = -\rho\overline{\mathbf{v}'\mathbf{v}'} \tag{B.5}$$

yielding a tensor with nine components. However, due to symmetry of this tensor, the tensor actually contains only six independent components. The sum of the normal stresses (diagonal) defines the turbulent kinetic energy k ($k = \frac{1}{2}\overline{\mathbf{v}'\mathbf{v}'}$). So it can be seen that turbulence modeling consists of finding 6 additional relations in order to close the equations. One of the most significant approaches to handle turbulence modeling is the so called Boussinesq hypothesis. The idea is based on the observation that the momentum transfer in turbulent flow is dominated by mixing due to large energetic turbulent eddies. The hypothesis assumes that there is a linear relation between turbulent shear stress τ_{ij}^{R} and mean rate of strain $\overline{S}_{ij}$, as it is in laminar flow. The proportionality factor is called eddy viscosity μ_t. For Reynolds-averaged incompressible flow the relation can be written as

$$\tau_{ij}^{R} = -\rho\overline{\mathbf{v}'\mathbf{v}'} = 2\mu_t\overline{S}_{ij} - \frac{2}{3}\rho k\delta_{ij} \tag{B.6}$$

in which k denotes the turbulent kinetic energy and δ_{ij} is the Kronecker delta. This eddy-viscosity concept is very interesting from an engineering point, since it requires "only" the determination of μ_t (since k is either obtained as a by-product of the turbulence model or is omitted). Once eddy viscosity is known, turbulent flow can be simulated by introducing averaged flow variables and adding μ_t to laminar viscosity to the Navier-Stokes equations. This approach can be used for a lot of applications, but is no longer valid in flows in which the equilibrium between the turbulence and the mean strain field is no longer present (i.e. flows with rotation, flows with significant streamline curvature, flows with sudden change in mean strain rate, ...) [125].

B.2 Realizable $k - \epsilon$ Turbulence Model

The standard $k - \epsilon$ model was originally established by Launder and Spalding [71] to be suitable from a practical perspective. Other modifications to the two-equation model have produced the RNG and the Realizable $k - \epsilon$ models. In this work, the Realizable version is considered for implementaton. Details are retrieved from [126]. The two-equations k and ϵ are given by

$$\frac{\partial \rho k}{\partial t} + \nabla\left(\rho \mathbf{v} k\right) = \nabla\left[(\mu + \frac{\mu_t}{\sigma_k})\nabla k\right] + P_k - \rho\epsilon - Y_M + S_k \tag{B.7}$$

$$\frac{\partial \rho \epsilon}{\partial t} + \nabla\left(\rho \mathbf{v} \epsilon\right) = \nabla\left[(\mu + \frac{\mu_t}{\sigma_\epsilon})\nabla \epsilon\right] + \rho C_1 S\epsilon - \rho C_2 \frac{\epsilon^2}{k + sqrt(\nu\epsilon)} + S_\epsilon \tag{B.8}$$

where

$$C_1 = \max\left[0.43, \frac{\eta}{\eta + 5}\right], \qquad \eta = S\frac{k}{\epsilon}, \qquad S = \sqrt{2S_{ij}S_{ij}}$$

P_k is the generation of turbulence kinetic energy due to the mean velocity, and μ_t is the turbulent viscosity, given by

$$\mu_t = \rho C_\mu \frac{k^2}{\epsilon}$$

The parameters are defined as

$$\begin{aligned} C_\mu &= \frac{1}{A_0 + A_s\frac{kU^*}{\epsilon}} \\ U^* &= \sqrt{S_{ij}S_{ij} + \tilde{\Omega}_{ij}\tilde{\Omega}_{ij}} \\ \tilde{\Omega}_{ij} &= \Omega_{ij} - 2\epsilon_{ijk}\omega_k \\ \Omega_{ij} &= \overline{\Omega_{ij}} - \epsilon_{ijk}\omega_k \end{aligned}$$

and $\overline{\Omega_{ij}}$ is the mean rate of rotation tensor viewed in a rotating reference frame with the angular velocity ω_k. The model constants A_0 and A_s are given by

$$A_0 = 4.04, \qquad A_s = \sqrt{6}\cos\phi$$

$$\phi = \frac{1}{3}\cos^{-1}\left(\sqrt{6}W\right), \qquad W = \frac{S_{ij}S_{jk}S_{ki}}{\tilde{S}_3}, \tilde{S} = \sqrt{S_{ij}S_{ij}}, \qquad S_{ij} = \frac{1}{2}\left(\nabla \mathbf{v} + \nabla \mathbf{v}^T\right)$$

The model constants are

$$C_{1\epsilon} = 1.44, \qquad C_2 = 1.9, \qquad \sigma_k = 1.0, \qquad \sigma_\epsilon = 1.2$$

B.3 $k-\omega$ SST Turbulence Model

The $k-\omega$ SST turbulence model [127] is a two-equation eddy-viscosity model which has become very popular. The shear stress transport (SST) formulation combines the best of two worlds. The use of a $k-\omega$ formulation in the inner parts of the boundary layer makes the model directly usable all the way down to the wall through the viscous sub-layer, hence the SST $k-\omega$ model can be used as a Low-Re turbulence model without any extra damping functions. The SST formulation also switches to a $k-\epsilon$ behavior in the free-stream and thereby avoids the common $k-\omega$ problem that the model is too sensitive to the inlet free-stream turbulence properties. Authors who use the SST $k-\omega$ model often merit it for its good behavior in adverse pressure gradients and separating flow. The SST $k-\omega$ model does produce a bit too large turbulence levels in regions with large normal strain, like stagnation regions and regions with strong acceleration. This tendency is much less pronounced than with a normal $k-\epsilon$ model though.

$$\frac{\partial \rho k}{\partial t} + \nabla(\rho \mathbf{v} k) = 2\mu_t - \rho\epsilon + \nabla\left[\frac{\mu_t}{\sigma_k}\nabla k\right] \tag{B.9}$$

$$\frac{\partial \rho \omega}{\partial t} + \nabla(\rho \mathbf{v} \omega) = \frac{\rho\gamma\overline{\tau}_{ij}\nabla \mathbf{v}}{\mu_t} - \beta\rho\omega^2 + \nabla\left[(\mu + \sigma_\omega \mu_t)\nabla\omega\right] + 2(1-F_1)\frac{\rho\sigma_{\omega 2}}{\omega}\nabla k \nabla\omega \tag{B.10}$$

The turbulent viscosity μ_t is computed from:

$$\mu_t = \frac{\rho a_1 k}{\max(a_1\omega, \Omega F_2)}$$

Each of the constants is a blend of an inner (1) and outer (2) constant, blended via:

$$\phi = F_1\phi_1 + (1-F_1)\phi_2$$

where ϕ_1 represents constant 1 and ϕ_2 represents constant 2. Additional functions are given by:

$$\begin{aligned}
F_1 &= \tanh(arg_1^4) \\
arg_1 &= \min\left[\max\left(\frac{\sqrt{k}}{\beta^*\omega d}, \frac{500\nu}{d^2\omega}\right), \frac{4\rho\sigma_{\omega 2}k}{CD_{k\omega}d^2}\right] \\
CD_{k\omega} &= \max\left(2\rho\sigma_{\omega 2}\frac{1}{\omega}\nabla k\nabla\omega, 10^{-20}\right) \\
F_2 &= \tanh(arg_2^2) \\
arg_2 &= \max\left(2\frac{\sqrt{k}}{\beta^*\omega d}, \frac{500\nu}{d^2\omega}\right)
\end{aligned}$$

and d is the distance from the field point to the nearest wall, and $\Omega = \sqrt{2W_{ij}W_{ij}}$ is the vorticity magnitude, with

$$W_{ij} = \frac{1}{2}\left(\nabla \mathbf{v} - \nabla \mathbf{v}^T\right)$$

The constants are:

$$\begin{aligned} \gamma_1 = \frac{\beta_1}{\beta^*} - \frac{\sigma_{\omega 1}\kappa^2}{\sqrt{\beta^*}}, \quad \gamma_1 = \frac{\beta_1}{\beta^*} - \frac{\sigma_{\omega 1}\kappa^2}{\sqrt{\beta^*}} \\ \sigma_{k1} = 0.85, \quad \sigma\omega 1 = 0.5, \quad \beta_1 = 0.075 \\ \sigma_{k2} = 1.0, \quad \sigma\omega 2 = 0.856, \quad \beta_2 = 0.0828 \\ \beta^* = 0.09, \quad \kappa = 0.51, \quad a_1 = 0.31 \end{aligned}$$

B.4 Transition SST Turbulence Model

This model is an extension to the SST model which is based on the Boussinesq eddy viscosity assumption. In addition to the original k and ω equations, two additional equations are added for the intermittency γ and the Reynolds number based on thickness onset transition $\hat{Re}_{\theta t}$. The effects of laminar-turbulent transition which are provided by the additional two equations are introduced to the SST model by modifying the production and destruction terms in the turbulent kinetic energy equation k. The 4 equations are presented here, and the rest of details are presented in the appendix:

$$\begin{aligned} \frac{\partial \rho k}{\partial t} + \nabla(\rho \mathbf{v} k) &= \gamma_{eff}\overline{\tau}_{ij}\nabla \mathbf{v} - \min(\max(\gamma_{eff}, 0.1), 1.0)\beta^* \rho\omega k + \nabla\left[(\mu + \sigma_k \mu_t)\nabla k\right] \\ \frac{\partial \rho \omega}{\partial t} + \nabla(\rho \mathbf{v} \omega) &= \frac{\rho\gamma\overline{\tau}_{ij}\nabla \mathbf{v}}{\mu_t} - \beta\rho\omega^2 + \nabla\left[(\mu + \sigma_\omega \mu_t)\nabla \omega\right] + 2(1 - F_1)\frac{\rho\sigma_{\omega 2}}{\omega}\nabla k \nabla \omega \\ \frac{\partial \rho \gamma}{\partial t} + \nabla(\rho \mathbf{v} \gamma) &= F_{length} c_{a1} \rho S[\gamma F_{onset}]^{0.5}(1 - c_{e1}\gamma) - c_{a2}\rho\Omega\gamma F_{turb}(c_{e2}\gamma - 1)\nabla\left[(\mu + \frac{\mu_t}{\sigma_f})\nabla\gamma\right] \\ \frac{\partial \rho \hat{Re}_{\theta t}}{\partial t} + \nabla(\rho \mathbf{v} \hat{Re}_{\theta t}) &= c_{thetat}\frac{\rho}{T}(Re_{\theta t}^{eq} - \hat{Re}_{\theta t})(1.0 - F_{\theta t}) + \nabla\left[\sigma_{\theta t}(\mu + \mu_t)\nabla \hat{Re}_{\theta t}\right] \end{aligned}$$

where

$$
\begin{aligned}
F_{onset1} &= \frac{Re_V}{2.193 Re_{\theta c}} \\
Re_V &= \frac{\rho S d^2}{\mu} \\
F_{onset2} &= \min(\max(F_{onset1}, F_{onset1}^4), 2.0) \\
R_T &= \frac{\rho k}{\mu \omega} \\
F_{onset3} &= \max\left[1 - \left(\frac{R_T}{2.5}\right)^3, 0\right] \\
F_{onset} &= \max(F_{onset2} - F_{onset3}, 0) \\
F_{turb} &= exp\left[-\left(\frac{R_T}{4}\right)^4\right] \\
F_{length} &= F_{length,1}(1 - F_{sublayer}) + 40.0 F_{sublayer}
\end{aligned}
$$

$$
F_{length,1} = \begin{cases}
39.8189 + (-119.270 \times 10^{-4})\hat{R}e_{\theta t} + (-132.567 \times 10^{-6})\hat{R}e_{\theta t}^2, & \hat{R}e_{\theta t} < 400; \\
363.404 + (-123.939 \times 10^{-2})\hat{R}e_{\theta t} + (194.548 \times 10^{-5})\hat{R}e_{\theta t}^2 & \\
\qquad + (-101.695 \times 10^{-8})\hat{R}e_{\theta t}^3, & 400 \leq \hat{R}e_{\theta t} < 596; \\
0.5 - (3.0 \times 10^{-4})(\hat{R}e_{\theta t} - 596.0), & 596 \leq \hat{R}e_{\theta t} < 1200; \\
0.3188 & 1200 \leq \hat{R}e_{\theta t}
\end{cases}
$$

$$
\begin{aligned}
F_{sublayer} &= exp\left[-\left(\frac{Re_\omega}{200}\right)^2\right] \\
Re_\omega &= \frac{\rho \omega d^2}{\mu}
\end{aligned}
$$

$$
Re_{\theta c} = \begin{cases}
(-396.035 \times 10^{-2}) + (10120.656 \times 10^{-4})\hat{R}e_{\theta t} + (-868.230 \times 10^{-6})\hat{R}e_{\theta t}^2 & \\
\qquad + (696.506 \times 10^{-9})\hat{R}e_{\theta t}^3 + (-174.105 \times 10^{-12})\hat{R}e_{\theta t}^4, & \hat{R}e_{\theta t} \leq 1870; \\
\hat{R}e_{\theta t} - (593.11 + 0.482(\hat{R}e_{\theta t} - 1870.0)), & 1870 < \hat{R}e_{\theta t}
\end{cases}
$$

The functions and parameters in the $\hat{R}e_{\theta t}$ equation are:

$$
\begin{aligned}
T &= \frac{500\mu}{\rho U^2} \\
F_{\theta t} &= \min\left[\max\left(F_{wake} exp\left(-\left(\frac{d}{\delta}\right)^4\right), 1.0 - \left(\frac{c_{e2}\gamma - 1}{c_{e2} - 1}\right)^2\right), 1.0\right] \\
\delta &= \frac{375\Omega\mu\hat{R}e_{\theta t} d}{\rho U^2} \\
F_{wake} &= exp\left[-\left(\frac{Re_\omega}{1 \times 10^5}\right)^2\right] \\
\lambda_\theta &= \frac{\rho\theta_t^2}{\mu}\frac{dU}{ds} \\
Tu &= 100\frac{\sqrt{2k/3}}{U} \\
\frac{dU}{ds} &= \frac{u_m u_n}{U^2}\frac{\partial u_m}{\partial x_n}
\end{aligned}
$$

$$
Re_{\theta t}^{eq} = \begin{cases} (1173.51 - 589.428Tu + 0.2196Tu^{-2})F(\lambda_\theta), & Tu \leq 1.3; \\ 331.5(Tu - 0.5658)^{-0.671}F(\lambda_\theta), & Tu > 1.3 \end{cases}
$$

$$
F(\lambda_\theta) = \begin{cases} 1 + [12.986\lambda_\theta^2 + 405.689\lambda_\theta^3] exp\left(-(\frac{Tu}{1.5})^{1.5}\right), & \lambda_\theta \leq 0; \\ 1 + 0.275[1 - exp(-35.0\lambda_\theta)] exp(-\frac{Tu}{0.5}), & \lambda_\theta > 0 \end{cases}
$$

The effects of laminar-turbulent transition are introduced to the underlying SST model by modifying the turbulent-kinetic-energy source terms as:

$$
\begin{aligned}
\gamma_{eff} &= \max(\gamma, \gamma_{sep}) \\
\gamma_{sep} &= \min\left(s_1 \max\left[0, \left(\frac{Re_V}{3.235Re_{\theta c}}\right) - 1\right] F_{reattach}, 2\right) \\
F_{reattach} &= exp\left[-\left(\frac{R_T}{20}\right)^4\right]
\end{aligned}
$$

The calibration constants for the Langtry-Menter model are:

$$\begin{aligned}
c_{a1} &= 2.0 \\
c_{a2} &= 0.06 \\
c_{e1} &= 1.0 \\
c_{e2} &= 50 \\
c_{\theta t} &= 0.03 \\
s_1 &= 2 \\
\sigma_f &= 1.0 \\
\sigma_{\theta t} &= 2.0
\end{aligned}$$

B.5 EARSM Turbulence Model

The EARSM enables an extension of the current ($k - \epsilon$ and BSL) turbulence models to capture secondary flows and flows with streamline curvature and system rotation. The implementation is based on the explicit algebraic Reynolds stress model of Wallin and Johansson [33]. The Reynolds stresses are computed from the anisotropy tensor according to its definition:

$$\overline{u_i u_j} = k \left(a_{ij} + \frac{2}{3}\delta_{ij} \right)$$

where the anisotropy tensor is searched as a solution of the following implicit algebraic matrix equation:

$$N\mathbf{a} = -A_1\mathbf{S} + (\mathbf{a\Omega} - \mathbf{\Omega a}) - A_2 \left(\mathbf{aS} - \mathbf{Sa} - \frac{2}{3} tr(\mathbf{aS}) \right)$$

with $N = A_3 + A_4 \left(\frac{P_k}{\epsilon}\right)$. The coefficients A_i in this matrix equation depend on the C_i-coefficients of the pressure-strain term in the underlying Reynolds stress transport model. Their values are selected here as $A_1 = 1.245$, $A_2 = 0$, $A_3 = 1.8$, $A_4 = 2.25$. S_{ij} and Ω_{ij} denote the non-dimensional strain-rate and vorticity tensors, respectively. They are defined as:

$$S_{ij} = \frac{1}{2}\tau \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right)$$

$$\Omega_{ij} = \frac{1}{2}\tau \left(\frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i} \right)$$

where the time-scale τ is given by:

$$\tau = \frac{k}{\epsilon}$$

In order to arrive at an explicit solution of the eq. B.5, the anisotropy tensor is expressed as a polynomial based on the strain rate and the vorticity tensors as follows:

$$\begin{aligned} a_{ij} &= \beta_1 S_{ij} + \beta_3 \left(\Omega_{ik}\Omega_{kj} - \frac{1}{3} II_\Omega \delta_{ij} \right) + \beta_4 \left(S_{ik}\Omega_{kj} - \Omega_{ik}S_{kj} \right) \\ &+ \beta_6 \left(S_{ik}\Omega_{kl}\Omega_{lj} + \Omega_{ik}\Omega_{kl}S_{lj} - \frac{2}{3} IV \delta_{ij} - II_\Omega S_{ij} \right) \end{aligned}$$

The β-coefficients are evaluated to:

$$\begin{aligned} \beta_1 &= -N/Q \\ \beta_3 &= -12\ IV/\left(NQ(2N^2 - II_\Omega)\right) \\ \beta_4 &= -1/Q \\ \beta_6 &= -6N/\left(Q(2N^2 - II_\Omega)\right) \end{aligned}$$

where the denominator Q is:

$$Q = \left(N^2 - 2\ II_\Omega\right)/A_1$$

The invariants, which appear in the formulation of the anisotropy tensor and the coefficients, are defined by:

$$\begin{aligned} II_S &= S_{kl}S_{lk} \\ II_\Omega &= \Omega_{kl}\Omega_{lk} \\ IV &= S_{kl}\Omega_{lm}\Omega_{mk} \end{aligned}$$

In three-dimensional flows, the equation to be solved for the function N is of sixth order and no explicit solution can be derived, whereas in two-dimensional mean flows the function can be derived from a cubic equation, an analytic solution of which is recommended by Wallin and Johansson [33] also for three-dimensional cases:

$$N = \begin{cases} A_3/3 + \left(P_1 + \sqrt{P_2}\right)^{1/3} + sign\left(P_1 - \sqrt{P_2}\right) \left| P_1 - \sqrt{P_2} \right|^{1/3} & \text{for } P_2 \geq 0 \\ A_3/3 + 2\left(P_1^2 - P_2\right)^{1/6} \cos\left(\frac{1}{3} \arccos\left(\frac{P_1}{\sqrt{P_1^2 - P_2}} \right) \right) & \text{for } P_2 < 0 \end{cases}$$

where

$$\begin{aligned} P_1 &= \left(\frac{A_3}{27} + \frac{A_1 A_4}{6}\ II_S - \frac{2}{3} II_\Omega \right) A_3 \\ P_2 &= P_1^2 - \left(\frac{A_3^2}{9} + \frac{A_1 A_4}{3}\ II_S + \frac{2}{3} II_\Omega \right)^3 \end{aligned}$$

Eventually, the eddy viscosity is calculated as follows:

$$\mu_t = C_\mu k^2/\epsilon$$

Appendix C

Linearization Strategy for Coupling Terms: Picard vs. Newton-Raphson

C.1 Overview

In this section, linearization strategies are considered. Nonlinearities are frequently encountered in fluid flow equations. Coupling strategies usually evolve from a linearization procedure to a nonlinear term of a product form i.e. (xy) in the equations. There are two main linearization methods, Picard and Newton-Raphson. The next sections address their concepts. A sample test of arbitrary nonlinear system will be tested to prove the superiority of Newton-Raphson linearization technique. The arbitrary system to be considered here is:

$$\begin{aligned} xy + y &= -1 \\ x + y &= 0 \end{aligned} \tag{C.1}$$

Solving the above system using an iterative procedure requires a sort of linearization to the nonlinear term xy in the first equation. Considering the two alternatives mentioned above, the convergence of each is tracked. The general algorithm to solve it is presented in algorithm 4.

C.2 Convergence Test

Considering a Picard linearization strategy, the system which exhibits a nonlinear term that couples two variables ϕ and ψ may be linearized such that:

$$\phi^{(n)}\psi^{(n)} = \phi^{(n-1)}\psi^{(n)} \tag{C.2}$$

Algorithm 4 Solve nonlinear system incorporating a linearization method

```
initialize $x$ and $y$
while $res > tol$ do
    solve linear system for $x^{(n)}$ and $y^{(n)}$
    if $n > maxIter$ then
        stop
    else
        update $res$
        $n = n + 1$
    end if
end while
```

where n is the index of the current iteration within the iterative process. Then, the system above (system C.1) can then be linearized as follows:

$$\begin{aligned} y^{(n-1)}x^{(n)} + y^{(n)} &= -1 \\ x^{(n)} + y^{(n)} &= 0 \end{aligned} \tag{C.3}$$

Newton-Raphson linearization applies the following to the nonlinear term $\phi\psi$:

$$\phi^{(n)}\psi^{(n)} = \phi^{(n-1)}\psi^{(n)} + \phi^{(n)}\psi^{(n-1)} - \phi^{(n-1)}\psi^{(n-1)} \tag{C.4}$$

The corresponding linear system at iter n is:

$$\begin{aligned} y^{(n-1)}x^{(n)} + (x^{(n-1)} + 1)y^{(n)} &= x^{(n-1)}y^{(n-1)} - 1 \\ x^{(n)} + y^{(n)} &= 0 \end{aligned} \tag{C.5}$$

These two linearizations are applied and their corresponding systems are solved accordingly. Starting by the following initial guess:

$$x = 2; \quad y = -1 \tag{C.6}$$

The residuals history for the linearized systems is plotted in the figure below: This test gives a sense of superiority of the Newton-Raphson approach over the conventional Picard linearization. This is the basis of the linearization technique used in linearizing the advection term in the α^k equations to be coupled to the velocity and pressure equations.

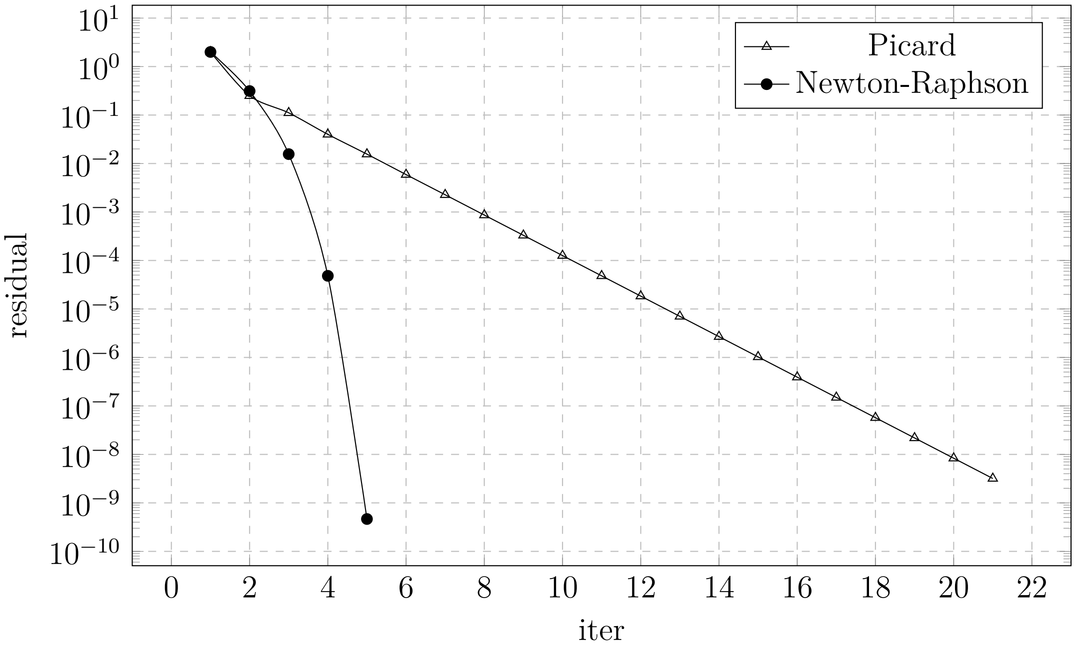


Figure C.1: Residuals history for Picard and Newton-Raphson linearization techniques

Appendix D

Schemes and Methods

D.1 Advection Schemes

D.1.1 Interface Capturing Schemes

An appropriate interface capturing scheme is to be used in advecting the volume fraction field. One way to address this objective is a switching strategy that depends on the angle between the flow direction and the grid lines, and this is approached by a continuous switching in which compressive and a high resolution advection scheme are blended together based on the aforementioned angle. To cite few, CICSAM [14], HRIC [15] and STACS [3] utilize this strategy.

HRIC Scheme

The current scheme combines the compressive property of a bounded downwind scheme (BD) and the stability of an upwind scheme. The normalized variable formulation of the BD scheme is:

$$\tilde{\alpha}^k_{f,BD} = \begin{cases} 2\tilde{\alpha}^k_C & 0 < \tilde{\alpha}^k_C \leq 0.5 \\ 1 & 0.5 < \tilde{\alpha}^k_C \leq 1 \\ \tilde{\alpha}^k_C & otherwise \end{cases} \tag{D.1}$$

A blending which occurs between the BD scheme and the upwind scheme is a function of the angle θ_f at the element face. θ_f is determined from the gradient of α^k:

$$\theta_f = \arccos \frac{\nabla \alpha^k_f \cdot \mathbf{d}_{CF}}{||\nabla \alpha^k_f||\ ||\mathbf{d}_{CF}||} \tag{D.2}$$

Then, $\tilde{\alpha}^k_f(\theta_f)$ is expressed such that:

$$\tilde{\alpha}^k_f(\theta_f) = \tilde{\alpha}^k_{f,BD}\sqrt{\cos\theta_f} + \tilde{\alpha}^k_{f,UPWIND}\left(1 - \sqrt{\cos\theta_f}\right) \tag{D.3}$$

A modification to the HRIC scheme to account for the local Courant number Co_f:

$$\tilde{\alpha}_f^k = \tilde{\alpha}_f^k(\theta_f) + \left(\tilde{\alpha}_f^k(\theta_f) - \tilde{\alpha}_{f,UPWIND}^k\right) \frac{0.7 - Co_f}{0.7 - 0.3} \tag{D.4}$$

and Co_f is calculated as:

$$Co_f = \frac{|\mathbf{v}_f \cdot \mathbf{S}_f| \Delta t}{V_f} \tag{D.5}$$

CICSAM Scheme

The CICSAM scheme switches between the HYPER-C scheme, which is a bounded downwind scheme, and the ULTIMATE-QUICKEST scheme (UQ). The HYPER-C enforces the CBC criterion onto the downwind scheme:

$$\tilde{\alpha}_{f,HYPER-C}^k = \begin{cases} \min\left(1, \frac{\tilde{\alpha}_C^k}{Co_f}\right) & 0 \leq \tilde{\alpha}_C^k \leq 1 \\ \tilde{\alpha}_C^k & otherwise \end{cases} \tag{D.6}$$

and Co_f is calculated as:

$$Co_f = \frac{|\mathbf{v}_f \cdot \mathbf{S}_f| \Delta t}{V_f} \tag{D.7}$$

And regarding the UQ scheme, its normalized variable formulation:

$$\tilde{\alpha}_{f,UQ}^k = Co_f \tilde{\alpha}_{f,UPWIND}^k + (1 - Co_f)\, \tilde{\alpha}_{f,QUICK}^k \tag{D.8}$$

where

$$\tilde{\alpha}_{f,QUICK}^k = \frac{3}{8} + \frac{3}{4} \tilde{\alpha}_C^k \tag{D.9}$$

and

$$\tilde{\alpha}_{f,UPWIND}^k = \tilde{\alpha}_C^k \tag{D.10}$$

Therefore, the CICSAM scheme is mathematically expressed as follows:

$$\tilde{\alpha}_{f,CICSAM}^k = \tilde{\alpha}_{f,HYPER-C}^k f(\theta_f) + \tilde{\alpha}_{f,UQ}^k (1 - f(\theta_f)) \tag{D.11}$$

and $f(\theta_f)$ is determined as:

$$f(\theta_f) = \min\left(\frac{\cos(2\theta_f) + 1}{2}, 1\right) \tag{D.12}$$

with θ_f being:

$$\theta_f = \arccos \frac{\nabla \alpha_f^k \cdot \mathbf{d}_{CF}}{||\nabla \alpha_f^k||\ ||\mathbf{d}_{CF}||} \tag{D.13}$$

STACS Scheme

The scheme is shown to be more accurate and less diffusive, preserving interface sharpness and boundedness at all Courant number values. Using the blending strategy, the normalized volume fraction value α_f^k at a face f, is calculated as:

$$\tilde{\alpha}^k_{f,STACS} = \tilde{\alpha}^k_{f,SUPERBEE}(\cos\theta_f)^4 + \tilde{\alpha}^k_{f,STOIC}\left[1 - (\cos\theta_f)^4\right] \tag{D.14}$$

where $\tilde{\alpha}^k_{f,SUPERBEE}$ and $\tilde{\alpha}^k_{f,STOIC}$ are obtained from

$$\tilde{\alpha}^k_{f,SUPERBEE} = \begin{cases} \tilde{\alpha}^k_C & \tilde{\alpha}^k_C \leq 0 \\ 1 & 0 < \tilde{\alpha}_C < 1 \\ \tilde{\alpha}^k_C & 1 \leq \tilde{\alpha}^k_C \end{cases} \qquad \tilde{\alpha}_{f,STOIC} = \begin{cases} \tilde{\alpha}^k_C & \tilde{\alpha}^k_C \leq 0 \\ \frac{1}{2} + \frac{1}{2}\tilde{\alpha}^k_C & 0 < \tilde{\alpha}^k_C \leq \frac{1}{2} \\ \frac{3}{8} + \frac{3}{4}\tilde{\alpha}^k_C & \frac{1}{2} < \tilde{\alpha}^k_C \leq \frac{5}{6} \\ 1 & \frac{5}{6} < \tilde{\alpha}^k_C \leq 1 \\ \tilde{\alpha}^k_C & 1 \leq \tilde{\alpha}^k_C \end{cases} \tag{D.15}$$

and the angle θ_f is calculated as:

$$\theta_f = \arccos \frac{\nabla\alpha^k_f \cdot \mathbf{d}_{CF}}{||\nabla\alpha^k_f||\ ||\mathbf{d}_{CF}||} \tag{D.16}$$

The value of α^k_f is computed from the normalized formulation such as:

$$\alpha^k_f = \tilde{\alpha}^k_f(\alpha^k_D - \alpha^k_U) + \alpha^k_U \tag{D.17}$$

where α^k_U and α^k_D are the upwind and downwind values of α, noting that α_U is actually reconstructed for unstructured non-orthogonal meshes such as shown in the following formula:

$$\alpha^k_U = \alpha^k_D - 2(\nabla\alpha^k_C \cdot \mathbf{d}_{CF}) \tag{D.18}$$

D.1.2 High Resolution Schemes

Van Leer Scheme

The TVD formulation of the Van-Leer scheme is presented here. The general formula of the high resolution value at the face is expressed as:

$$\alpha^k_f = \alpha^k_C + \frac{1}{2}\psi(r_f)(\alpha^k_D - \alpha^k_C) \tag{D.19}$$

with

$$r_f = \frac{\alpha^k_C - \alpha^k_U}{\alpha^k_D - \alpha^k_C} \tag{D.20}$$

U is the node upwind to C, and D is the node downwind to C. The TVD-Van Leer limiter $\psi(r_f)$ is determined to be:

$$\psi(r_f) = \frac{r_f + |r_f|}{1 + |r_f|} \tag{D.21}$$

Barth and Jesperson Linear Reconstruction Scheme

The interpolated volume fraction value to a face α_f^k is obtained with higher order accuracy by performing piece-wise linear reconstructions of the cell averaged solution data while ensuring that new extrema are not created in the reconstruction process. The method is addressed by Barth and Jesperson [22] and it is both accurate and bounded, since it only reduces to first order near discontinuities. The method was modified by [128] to apply it for the reconstruction of α^k within the control volume. The High Resolution Scheme appears as:

$$\alpha_f^k = \alpha_C^k + \beta_C \nabla \alpha_C^k \cdot \mathbf{d}_{Cf} \tag{D.22}$$

where β_C is computed locally to be as close to some upper bound β_{max} as possible without introducing local oscillations. The recipe for β_C is based on the boundedness principles used by [22]. For all faces f of the element C:

$$\beta_f = \begin{cases} \min\left(\frac{\max(\phi_C, \phi_F) - \phi_C}{\nabla \phi_C \cdot \mathbf{d}_{Cf}}, \beta_{max}\right) & \phi_f > \phi_C \\ \min\left(\frac{\phi_C - \min(\phi_C, \phi_F)}{\nabla \phi_C \cdot \mathbf{d}_{Cf}}, \beta_{max}\right) & \phi_f < \phi_C \\ \beta_{max} & \phi_f = \phi_C \end{cases} \tag{D.23}$$

and

$$\beta_C = \min(\beta_f) \qquad \forall f \in nb(C) \tag{D.24}$$

The upper bound β_{max} for a typical high order performance is usually 1, which is very close to the performance of the FROMM scheme, however, setting β_{max} to 2 will shift the performance of the scheme towards a downwind behavior, that is, a compressive behavior. The latter behavior is desirable in free surface flows in order to maintain an acceptable sharpness for the interface.

D.2 Transient Schemes

For any transient behavior, the expression of time evolution of a variable ϕ is governed by the equation

$$\frac{\partial(\rho\phi)}{\partial t} + \mathcal{L}(t) = 0 \tag{D.25}$$

After applying a spatial discretization, the equation is of the form

$$\frac{\partial(\rho_C \phi_C)}{\partial t} V_C + L(\phi_C^t) = 0 \tag{D.26}$$

where V_C is the volume of the discretization element and $L(\phi_C^t)$ is the spatial discretization operator at some reference time t. There are various implicit transient schemes which lead to different formulations of the unsteady term. Selected implicit transient schemes are described next.

D.2.1 First Order Backward Euler

$$\frac{(\rho_C\phi_C)^t - (\rho_C\phi_C)^{t-\Delta t}}{\Delta t} V_C + L(\phi_C^t) = 0 \tag{D.27}$$

D.2.2 Second Order Backward Euler

$$\frac{3(\rho_C\phi_C)^t - 4(\rho_C\phi_C)^{t-\Delta t} + (\rho_C\phi_C)^{t-2\Delta t}}{2\Delta t} V_C + L(\phi_C^t) = 0 \tag{D.28}$$

D.3 Enhancement of Matrix Diagonal

D.3.1 Pseudo Transience for Steady-State Cases

Pseudo transience, called also false transience, is a special under-relaxation technique which aims at stabilizing a steady-state system. It also protects the algebraic system from an ill-posed state if no diffusion terms are involved. Considering the steady transport of a scalar ϕ, the general equation has the following form:

$$\nabla \cdot (\rho \mathbf{v} \phi) = \nabla \cdot (\Gamma^\phi \nabla \phi) + Q^\phi \tag{D.29}$$

The above equation can be under-relaxed by some relaxation factor to stabilize the solution progress of the corresponding discrete algebraic system. An alternative to this method, a pseudo-transience technique can be utilized instead. This is done by including a transient-like term which expresses in fact a false time marching from an iteration to another in the simulation. The transport equation of ϕ has now the form:

$$\frac{\partial \rho\phi}{\partial t} + \nabla \cdot (\rho \mathbf{v} \phi) = \nabla \cdot (\Gamma^\phi \nabla \phi) + Q^\phi \tag{D.30}$$

where t is the false transient time step. Applying the finite volume principles, including divergence theorem and midpoint integration rule, and others, the semi-discrete form of the equation over a finite volume C is:

$$\frac{(\rho_C\phi_C) - (\rho_C\phi_C)^{(n)}}{\Delta t} V_C + \sum_{f\sim nb(C)} \dot{m}_f \phi_f = \sum_{f\sim nb(C)} \Gamma_f^\phi \nabla\phi_f \cdot \mathbf{S}_f + Q_C^\phi V_C \tag{D.31}$$

The corresponding algebraic system is:

$$a_C\phi_C + \sum_{F\sim NB(C)} a_F\phi_F = b_C \tag{D.32}$$

where

$$
\begin{aligned}
a_C &= \frac{\rho_C^*}{\Delta t} V_C + \sum_{f \sim nb(C)} \left(||\dot{m}_f^*, 0|| + \Gamma_f^{\phi,*} \frac{E_f}{d_{CF}} \right) \\
a_F &= -|| - \dot{m}_f^*, 0|| - \Gamma_f^{\phi,*} \frac{E_f}{d_{CF}} \\
b_C &= \frac{(\rho_C \phi_C)^{(n)}}{\Delta t} V_C + \sum_{f \sim nb(C)} \Gamma_f^{\phi,*} \nabla \phi_f^* \cdot \mathbf{T}_f + Q_C^{\phi} V_C
\end{aligned}
$$

This strategy provides an enhanced stability to the simulation, and it is however crucial for cases that have weak or no diffusion processes, this is because convection terms contribute either to the diagonal or to the off-diagonal entries of the algebraic system depending on the direction of the flow. In the case where the convection term does not contribute to the diagonal, a zero diagonal could exist in the algebraic system, hence, the linear solver would fail in solving the system. Adding a false transient term to the conservation equation ensures a non-zero diagonal in the linear system.

D.3.2 Implicit Relaxation

Implicit-relaxation, or under-relaxation, is a diagonal enhancing method which increases the stability of the computation. Consider the algebraic system

$$
a_C \phi_C + \sum_{F \sim NB(C)} a_F \phi_F = b_C \tag{D.33}
$$

Re-arranging

$$
\phi_C = \frac{-\sum_{F \sim NB(C)} a_F \phi_F + b_C}{a_C} \tag{D.34}
$$

The scalar exact solution ϕ_C is relaxed by some relaxation factor λ^{ϕ} as

$$
\phi_C = \lambda^{\phi} \phi_C^{**} + (1 - \lambda^{\phi}) \phi_C^* \tag{D.35}
$$

where the double Asterisk $**$ corresponds to the new iteration value, and the single Asterisk $*$ as mentioned before is an old iteration value. Eq. D.35 can also be put in the form

$$
\phi_C = \phi_C^* + \lambda^{\phi} \left(\phi_C^{**} - \phi_C^* \right) \tag{D.36}
$$

The algebraic equation D.34 solves in fact for ϕ_C^{**}. Substituting D.36 in D.34

$$
\phi_C = \phi_C^* + \lambda^{\phi} \left[\left(\frac{-\sum_{F \sim NB(C)} a_F \phi_F + b_C}{a_C} \right) - \phi_C^* \right] \tag{D.37}
$$

the algebraic equation becomes

$$
\frac{a_C}{\lambda^{\phi}} \phi_C + \sum_{F \sim NB(C)} a_F \phi_F = b_C + \frac{1 - \lambda^{\phi}}{\lambda^{\phi}} a_C \phi_C^* \tag{D.38}
$$

D.3.3 Explicit Relaxation

The explicit relaxation of a field is a simple explicit formula where the field is relaxed after the solution of the equation

$$\phi_C^{***} = \lambda^\phi \phi_C^{**} + (1 - \lambda^\phi)\phi_C^* \tag{D.39}$$

where the triple Asterisk $***$ corresponds to the relaxed field.

D.3.4 Local Time Scale

A local time scale is a form of implicit relaxation using different input terminology. A local time scale for the equation of ϕ is referred to as τ^ϕ and it can be simply formulated in terms of an under relaxation λ^ϕ

$$\tau^\phi = \frac{\lambda^\phi}{1 - \lambda^\phi} \tag{D.40}$$

So without any further details, the algebraic equation would look like

$$\frac{1 + \tau^\phi}{\tau^\phi} a_C \phi_C + \sum_{F \sim NB(C)} a_F \phi_F = b_C + \frac{a_C}{\tau^\phi} \phi_C^* \tag{D.41}$$

D.3.5 Enforcement of Diagonal Dominance in the Pressure Equation for Compressible Cases

For compressible flows, diagonal dominance may not be ensured for reasons related to a pressure-advection like effect appearing in the pressure equation (addressed previously in the chapter of compressible considerations). However, It is recommended to ensure that the diagonal of the discrete algebraic pressure equation is always dominant. If the discrete equation is assembled in a residual form, then altering the diagonal alone would not have any influence on the solution. The diagonal therefore can be checked as such

$$a_C^{pp} < -\sum_{F \sim NB(C)} a_F^{pp} \tag{D.42}$$

If this is the case, then the diagonal is substituted as

$$a_C^{pp} = -\sum_{F \sim NB(C)} a_F^{pp} \tag{D.43}$$

if and only if the off-diagonal coefficient $a_F^{pp} \leq 0$, otherwise it is zeroed such that $a_F^{pp} = 0$.

D.3.6 Divergence Correction

From Reynolds transport theorem, the flow of any physical quantity ϕ excluding surface and body sources is governed by the following equation:

$$\frac{\partial}{\partial t}\left[\int_{\Omega} \phi dV\right] = -\int_{\partial\Omega} \phi \mathbf{v} \cdot \mathbf{n} dS \tag{D.44}$$

Ω is the control volume of interest and $\partial\Omega$ is the control volume surface. Assuming Ω is constant with time, the first term can be expressed as:

$$\frac{\partial}{\partial t}\left[\int_{\Omega} \phi dV\right] = \int_{\Omega} \frac{\partial \phi}{\partial t} dV \tag{D.45}$$

And applying divergence theorem to the term involving surface integral:

$$\int_{\partial\Omega} \phi \mathbf{v} \cdot \mathbf{n} dS = \int_{\Omega} \nabla \cdot (\phi \mathbf{v}) dV \tag{D.46}$$

Substituting equations D.45 and D.46 in equation D.44:

$$\int_{\Omega} \frac{\partial \phi}{\partial t} dV = -\int_{\Omega} \nabla \cdot (\phi \mathbf{v}) dV \tag{D.47}$$

Collecting within the same integral:

$$\int_{\Omega} \left[\frac{\partial \phi}{\partial t} + \nabla \cdot (\phi \mathbf{v})\right] dV = 0 \tag{D.48}$$

which necessitates that the integrand be zero:

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (\phi \mathbf{v}) = 0 \tag{D.49}$$

The second term of the above equation can be splitted due to the ∇ operator:

$$\frac{\partial \phi}{\partial t} + \mathbf{v} \cdot \nabla \phi + \phi \nabla \cdot \mathbf{v} = 0 \tag{D.50}$$

and since $\nabla \cdot \mathbf{v}$ is zero, then:

$$\frac{\partial \phi}{\partial t} + \mathbf{v} \cdot \nabla \phi = 0 \tag{D.51}$$

This is the material derivative's definition:

$$\frac{D\phi}{Dt} = \frac{\partial \phi}{\partial t} + \mathbf{v} \cdot \nabla \phi = 0 \tag{D.52}$$

And the above equation is the conservative form of the conservation equation. This is an introduction to the following sections, addressing the divergence correction term due to non-conservative aspect of the conservation equation during the simulation process.

General Transient-Advection Equation

Following the above discussion, multiplying the last equation by ρ:

$$\rho \frac{D\phi}{Dt} = 0 \tag{D.53}$$

From the definition of the substantial derivative, the equation can be further expressed as:

$$\rho \frac{\partial \phi}{\partial t} + \rho \mathbf{v} \cdot \nabla \phi = 0 \tag{D.54}$$

Inserting ρ into the time and space derivative operators:

$$\frac{\partial \rho \phi}{\partial t} - \phi \frac{\partial \rho}{\partial t} + \nabla \cdot \rho \mathbf{v} \phi - \phi \nabla \cdot \rho \mathbf{v} = 0 \tag{D.55}$$

Collecting terms:

$$\frac{\partial \rho \phi}{\partial t} + \nabla \cdot \rho \mathbf{v} \phi - \phi \underbrace{\left[\frac{\partial \rho}{\partial t} + \nabla \cdot \rho \mathbf{v} \right]}_{continuity} = 0 \tag{D.56}$$

Volume Fraction Equation

The volume fraction equation of a phase is in conservative form originally, so in its essence it is not a regular conservation equation, and hence the divergence correction term stated above for a general conservation equation does not apply. However, Rider et al. [109] has mentioned the addition of the divergence correction term to the original form of the advection equation of the volume fraction field. α^k of some phase k in the mixture is advected by $\mathbf{v}$:

$$\frac{D\alpha^k}{Dt} = \frac{\partial \alpha^k}{\partial t} + \mathbf{v} \cdot \nabla \alpha^k = 0 \tag{D.57}$$

The above equation can be recast into a conservative form such as:

$$\frac{\partial \alpha^k}{\partial t} + \nabla \cdot (\mathbf{v} \alpha^k) - \alpha^k \underbrace{\nabla \cdot \mathbf{v}}_{divergence} = 0 \tag{D.58}$$

The third term will vanish eventually, but it may be maintained to aid in achieving better diagonal, if and only if it has a positive contribution to the diagonal, otherwise it has to be treated explicitly, or could be zeroed though. It has to be also pointed out here that the current volume fraction equation being solved in the current study is the multiphase version, which is the form that includes the phasic density (see eq. 3.6). To determine the equivalent divergence correction term for the multiphase version of the volume fraction equation, eq. D.58 is multiplied by the phasic density ρ^k:

$$\rho^k \frac{\partial \alpha^k}{\partial t} + \rho^k \nabla \cdot (\mathbf{v}\alpha^k) - \rho^k \alpha^k \underbrace{\nabla \cdot \mathbf{v}}_{divergence} = 0 \tag{D.59}$$

Reforming the equation so as to insert ρ^k into the time and space derivative operators:

$$\frac{\partial \rho^k \alpha^k}{\partial t} - \alpha^k \frac{\partial \rho^k}{\partial t} + \nabla \cdot (\rho^k \mathbf{v}\alpha^k) - \alpha^k \mathbf{v} \cdot \nabla \rho^k - \alpha^k \nabla \cdot (\rho^k \mathbf{v}) + \alpha^k \mathbf{v} \cdot \nabla \rho^k = 0 \tag{D.60}$$

Taking out repeated terms, and collecting the remaining:

$$\frac{\partial \rho^k \alpha^k}{\partial t} + \nabla \cdot (\rho^k \mathbf{v}\alpha^k) - \alpha^k \left[\frac{\partial \rho^k}{\partial t} + \nabla \cdot (\rho^k \mathbf{v}) \right] = 0 \tag{D.61}$$

Eq. D.61 is the phasic equation presented earlier (eq. 3.6) plus a phasic divergence correction term.

D.3.7 Local Time Stepping

Instead of a fixed time step, the time step may be set to change according to a certain criterion. The VOF method is originally a transient free surface model. However, there are plenty of free surface flow problems that are in nature steady, and not exhibiting transient effects. Open channel flow is an example of a steady-state free surface flow problem, where the flow's surface assumes a steady height over the ground. Attempting to solve a steady free surface flow using a transient solver, will unjustifiably consume lots of time and memory until an asymptotic state is reached. Also, a steady state simulation assumes a very large time step such that the final state is reached rapidly, but this is only subject to the fact that a restrictive false transience is applied to relax the progress of the simulation. A local time stepping method has the potential to restrict the false transience based on the local Courant number of each individual element of the mesh, and is more restrictive in the interface regions of the domain. Additional damping and smoothing for the time step is applied to avoid discontinuous time steps across neighboring elements [129]. For an element C, the false time step Δt_C is limited such that:

$$\frac{1}{\Delta t_C} = \max \left(\frac{1}{\Delta t_{max}}, \frac{1}{\Delta t_0} \frac{Co_C}{Co_{max}} \right) \tag{D.62}$$

where Δt_0 is the set false time step, and Δt_{max} is the maximum set time step which may not be exceeded at any of the mesh elements. Co_C is the local Courant number at the element calculated as:

$$Co_C = \frac{\Delta t_0}{2V_C} \sum_{f \sim nb(C)} |\mathbf{v}_f \cdot \mathbf{S}_f| \tag{D.63}$$

Further limiting of the local time step is done exclusively for the elements in the interface region according to the interface maximum Courant number $Co_{\alpha,max}$:

$$\frac{1}{\Delta t_C} = \max\left(\frac{1}{\Delta t_C}, \frac{\chi_C}{\Delta t_0}\frac{Co_C}{Co_{\alpha,max}}\right) \tag{D.64}$$

where χ_C is an indicator which tells if element C is inside the interface region:

$$\chi_C = \begin{cases} 1 & 0.01 < \overline{\alpha_C^k} < 0.99 \\ 0 & otherwise \end{cases} \tag{D.65}$$

and $\overline{\alpha_C^k}$ is face-weighted average for the field α^k at element C, such that:

$$\overline{\alpha_C^k} = \frac{\sum\limits_{f\sim nb(C)} S_f \alpha_f^k}{\sum\limits_{f\sim nb(C)} S_f} \tag{D.66}$$

Note that $S_f \equiv \|\mathbf{S}_f\|$ is the area of the face f. The above limiting of the local time step could produce a discontinuous time step field which might introduce instabilities to the system due to large conservation errors. So smoothing the local time step field is made to avoid the aforementioned problem. Smoothing can be done by different ways. A quick and reliable way is by interpolating the local time step field from element centers to element faces, and then interpolating back from element faces to element centers. Another way is by looping over the time steps of the neighboring elements of an element C to ensure the values in neighboring elements are at least a coefficient k_s (i.e. 0.05) of the element value. A last step to apply here is damping the time step in the whole domain by considering:

$$\frac{1}{\Delta t_C} = \max\left(\frac{1}{\Delta t_C}, (1-k_d)\frac{1}{\Delta t_0}\right) \tag{D.67}$$

where k_d is a damping coefficient chosen between 0 and 1. Kim et al. [129] have mentioned in their work some remarks regarding the local time stepping, that is it convenient to set the smoothing coefficient to a value between 0.4 and 0.8, while they fixed their damping coefficient to unity. Other recommendations by [76] state that the Courant number should not exceed the value of $Co_{max} = 0.2$.

D.4 Residual Normalization

The residual of the equation at any given iteration and for an arbitrary element C in the computational domain is denoted by R_C^ϕ and is defined as

$$R_C^\phi = b_C - a_C\phi_C^* - \sum_{F\sim NB(C)} a_F \phi_F^* \tag{D.68}$$

A root-mean-square residual for the whole domain is calculated as

$$R^{\phi}_{rms} = \sqrt{\frac{\sum\limits_{C}\left(R^{\phi}_{C}\right)^{2}}{N}} \tag{D.69}$$

where N is the number of elements in the domain. The residual at each element is commonly normalized in order to provide a better insight of the convergence rate of the simulation. So, a normalized residual at an element C is written as

$$R^{\phi}_{C,normalized} = \frac{b_C - a_C\phi^{*}_{C} - \sum_{F\sim NB(C)} a_F\phi^{*}_{F}}{\max\limits_{all\ elements}\left|a_C\phi_C\right|} \tag{D.70}$$

Therefore, the corresponding root-mean-square residual is

$$R^{\phi}_{rms,normalized} = \sqrt{\frac{\sum\limits_{C}\left(R^{\phi}_{C,normalized}\right)^{2}}{N}} \tag{D.71}$$

D.5 Residual Form of the Algebraic System

Considering again the algebraic equation of ϕ

$$a_C\phi_C + \sum_{F\sim NB(C)} a_F\phi_F = b_C \tag{D.72}$$

A residual form (or correction form) of the equation above is commonly used instead of the full form. This is because a correction of the solution is usually smaller in magnitude than the full solution. The exact solution of ϕ can be decomposed into an approximation solution and a correction

$$\phi = \phi^{*} + \phi' \tag{D.73}$$

So, substituting in the the algebraic equation of ϕ

$$a_C\left(\phi^{*}_{C} + \phi'_{C}\right) + \sum_{F\sim NB(C)} a_F\left(\phi^{*}_{F} + \phi'_{F}\right) = b_C \tag{D.74}$$

therefore the residual form of the algebraic equation of ϕ is

$$a_C\phi'_C + \sum_{F\sim NB(C)} a_F\phi'_F = b_C - a_C\phi^{*}_{C} - \sum_{F\sim NB(C)} a_F\phi^{*}_{F} \tag{D.75}$$

and from the definition of the residual in eq. D.68, the above equation appears as

$$a_C\phi'_C + \sum_{F\sim NB(C)} a_F\phi'_F = R^{\phi}_{C} \tag{D.76}$$

The current work utilizes the residual form of the algebraic system rather than the full form.

D.6 Interpolation Schemes

D.6.1 Linear

$$\phi_f = g_f \phi_C + (1 - g_f)\phi_F \tag{D.77}$$

where

$$g_f = \frac{V_F}{V_C + V_F} \tag{D.78}$$

or more consistently

$$g_f = \frac{-\mathbf{d}_{Ff} \cdot \mathbf{n}_f}{-\mathbf{d}_{Ff} \cdot \mathbf{n}_f + \mathbf{d}_{Cf} \cdot \mathbf{n}_f} \tag{D.79}$$

D.6.2 Arithmetic

This interpolation is a simple average of the owner and neighbour values:

$$\phi_f = \frac{1}{2}(\phi_C + \phi_F) \tag{D.80}$$

D.6.3 Harmonic

This interpolation utilizes a weighted average such that:

$$\phi_f = \frac{1}{\frac{g_f}{\phi_C} + \frac{1-g_f}{\phi_F}} \tag{D.81}$$

D.6.4 Density-Based Harmonic

Any quantity ϕ which could be a scalar, a vector, or a tensor, may be interpolated by a harmonic interpolation with weights being the mixture density field, such that:

$$\phi_f = \rho_f \left[g_f \frac{\phi_C}{\rho_C} + (1 - g_f)\frac{\phi_F}{\rho_F} \right] \tag{D.82}$$

where ρ_f is harmonically interpolated as described previously:

$$\rho_f = \frac{1}{\frac{g_f}{\rho_C} + \frac{1-g_f}{\rho_F}} \tag{D.83}$$

D.6.5 Upwind

$$\phi_f = \begin{cases} \phi_C & \dot{m}_f > 0 \\ \phi_F & \dot{m}_f \leq 0 \end{cases} \tag{D.84}$$

D.7 Gradient Computation

Considering a Green Gauss gradient formulation, $\nabla\phi_C$ over an element C is expressed as:

$$\nabla\phi_C = \frac{1}{V_C}\sum_{f\sim nb(C)} \phi_f \mathbf{S}_f \tag{D.85}$$

ϕ_f is an interpolated value of ϕ at the face. Two methods for determining ϕ_f here, using a cell-based interpolation or a node-based one.

D.7.1 Cell-Based Green Gauss Gradient

Assuming a linear interpolation, ϕ_f is determined as:

$$\phi_f = g_f\phi_C + (1-g_f)\phi_F \tag{D.86}$$

D.7.2 Node-Based Green Gauss Gradient

ϕ_f is determined from the node values such that:

$$\phi_f = \frac{\sum\limits_{k=1}^{nb(f)} \frac{\phi_{n_k}}{\|\mathbf{r}_n - \mathbf{r}_{n_k}\|}}{\sum\limits_{k=1}^{nb(f)} \frac{1}{\|\mathbf{r}_n - \mathbf{r}_{n_k}\|}} \tag{D.87}$$

and the value of the node is interpolated from the cells neighboring the node:

$$\phi_n = \frac{\sum\limits_{k=1}^{NB(n)} \frac{\phi_{F_k}}{\|\mathbf{r}_n - \mathbf{r}_{F_k}\|}}{\sum\limits_{k=1}^{NB(n)} \frac{1}{\|\mathbf{r}_n - \mathbf{r}_{F_k}\|}} \tag{D.88}$$

D.7.3 Corrected Gradient at Face

At a face f, it is very common to calculate an improved face gradient by correcting the linearly interpolated gradient $\overline{\nabla\phi_f}$ by means of enhancing the component of the gradient in the direction of the vector connecting the centers of elements C and F that are straddling the face f:

$$\nabla\phi_f = \overline{\nabla\phi_f} + \left(\frac{\phi_F - \phi_C}{\|\mathbf{d}_{CF}\|^2}\mathbf{d}_{CF} - \overline{\nabla\phi_f}\cdot\frac{\mathbf{d}_{CF}}{\|\mathbf{d}_{CF}\|}\right) \tag{D.89}$$

D.7.4 Limiting the Gradient

Limiting the gradient is necessary to satisfy the monotonicty condition which prohibits the overshoot of the interpolated value of the field to the cell face. There are two gradient limiters, standard and multi-directional.

Standard

This approach is demonstrated in figure D.1. The gradient at element C is scaled such that the interpolated field value ϕ_f does not exceed the maximum value or drop below the minimum one. Denoting the limiter of the gradient at element C by β_C, it is defined as:

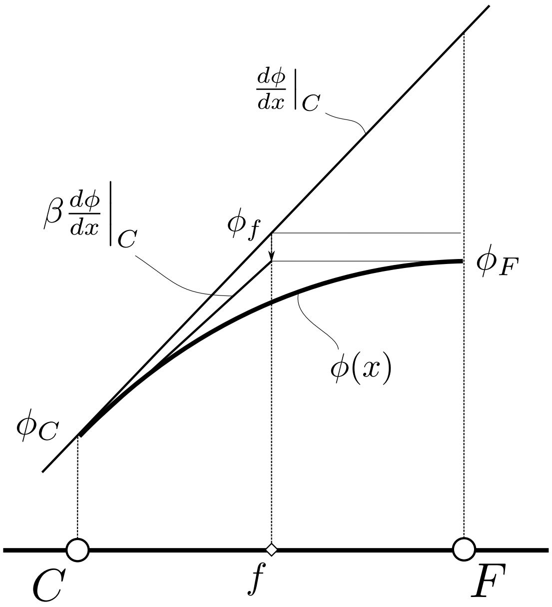


Figure D.1: One-dimensional standard gradient limiting demonstration. The gradient is scaled so as to clip the interpolated field at the face to the maximum value

$$\beta_C = \begin{cases} \underset{f\sim nb(C)}{\forall} \min\left(\frac{\phi_{C,max}}{\nabla\phi_C\cdot\mathbf{d}_{Cf}}, 1\right) & \nabla\phi_C\cdot\mathbf{d}_{Cf} > 0 \\ \underset{f\sim nb(C)}{\forall} \min\left(\frac{\phi_{C,min}}{\nabla\phi_C\cdot\mathbf{d}_{Cf}}, 1\right) & \nabla\phi_C\cdot\mathbf{d}_{Cf} < 0 \end{cases} \tag{D.90}$$

where $\phi_{C,max}$ and $\phi_{C,min}$ are defined as:

$$\begin{aligned} \phi_{C,max} &= \underset{F\sim NB(C)}{\forall} \max(\phi_C, \phi_F) - \phi_C \\ \phi_{C,min} &= \underset{F\sim NB(C)}{\forall} \min(\phi_C, \phi_F) - \phi_C \end{aligned} \tag{D.91}$$

Therefore, the gradient at element C is scaled by the limiter $beta_C$:

$$\nabla\phi_C := \beta_C\nabla\phi_C \tag{D.92}$$

Multi-Directional

The standard gradient limiter is severe in the sense that it clips the gradient in all directions in the same manner, even if the clipping is not required in a certain direction. The multi-directional limiter [130] considers this dilemma and applies the standard limiter but with clipping only the normal components of the gradient to the faces of the cell. Thus, the gradient at element C is adjusted as follows:

$$\nabla\phi_C := \begin{cases} \nabla\phi_C + \mathbf{d}_{Cf}\frac{(\phi_{C,max} - \nabla\phi_C \cdot \mathbf{d}_{Cf})}{\|\mathbf{d}_{Cf}\|^2} & \nabla\phi_C \cdot \mathbf{d}_{Cf} > \phi_{C,max} \\ \nabla\phi_C + \mathbf{d}_{Cf}\frac{(\phi_{C,min} - \nabla\phi_C \cdot \mathbf{d}_{Cf})}{\|\mathbf{d}_{Cf}\|^2} & \nabla\phi_C \cdot \mathbf{d}_{Cf} < \phi_{C,min} \end{cases} \tag{D.93}$$

where $\phi_{C,max}$ and $\phi_{C,min}$ are defined as:

$$\begin{aligned} \phi_{C,max} &= \underset{F\sim NB(C)}{\forall} \max(\phi_C, \phi_F) - \phi_C \\ \phi_{C,min} &= \underset{F\sim NB(C)}{\forall} \min(\phi_C, \phi_F) - \phi_C \end{aligned} \tag{D.94}$$

Appendix E

More on Finite Volume Method

E.1 Rhie-Chow Treatment due to Under-Relaxation and Transient Effects

An additional treatment related to the under-relaxation of the momentum equation used in conjunction with the SIMPLE algorithm for collocated grids is necessary. Accounting for the under-relaxation of the momentum equation in the Rhie-Chow interpolation eliminates its effect on the solution. Moreover, it is also necessary to account for the transient term effect on the face velocity. Disregarding this effect gives rise to oscillatory behavior of the solution for small time steps. Therefore, accounting for the above effects of under-relaxation and transient effect appears in the Rhie-Chow term as follows:

$$\mathbf{v}_f = \overline{\mathbf{v}_f} - \overline{\mathbf{D}_f^{\mathbf{v}}}(\nabla p_f - \overline{\nabla p_f}) + (1-\lambda^{\mathbf{v}})\left(\mathbf{v}_f^{(n)} - \overline{\mathbf{v}_f^{(n)}}\right) + \frac{\overline{a_f^{\circ}\mathbf{D}_f^{\mathbf{v}}}}{V_f}\left(\mathbf{v}_f^{\circ} - \overline{\mathbf{v}_f^{\circ}}\right) \tag{E.1}$$

where the superscript n refers to the previous iteration value. Note that the treatment related to the transient term is consistent with a backward Euler transient scheme. For other transient schemes, a different formulation for the treatment in the Rhie-Chow interpolation must be developed. The overall Rhie-Chow interpolation of the velocity to the cell face including the effects of body force, under-relaxation and transient effects is

$$\begin{aligned}\mathbf{v}_f = {} & \overline{\mathbf{v}_f} - \overline{\mathbf{D}_f^{\mathbf{v}}}(\nabla p_f - \overline{\nabla p_f}) + \overline{\mathbf{D}_f^{\mathbf{v}}}\left(\overline{\mathbf{B}_f} - \overline{\overline{\overline{\mathbf{B}_f}}}\right) \\ & + (1-\lambda^{\mathbf{v}})\left(\mathbf{v}_f^{(n)} - \overline{\mathbf{v}_f^{(n)}}\right) + \frac{\overline{a_f^{\circ}\mathbf{D}_f^{\mathbf{v}}}}{V_f}\left(\mathbf{v}_f^{\circ} - \overline{\mathbf{v}_f^{\circ}}\right)\end{aligned} \tag{E.2}$$

Finally, the mass flow rate $\dot{m}_f$ is calculated in the same way, i.e.,

$$\begin{aligned}\dot{m}_f = {} & \rho_f \overline{\mathbf{v}_f} \cdot \mathbf{S}_f - \rho_f \overline{\mathbf{D}_f^{\mathbf{v}}} (\nabla p_f - \overline{\nabla p_f}) \cdot \mathbf{S}_f + \rho_f \overline{\mathbf{D}_f^{\mathbf{v}}} \left(\overline{\mathbf{B}_f} - \overline{\overline{\overline{\mathbf{B}_f}}} \right) \cdot \mathbf{S}_f \\ & + \rho_f (1 - \lambda^{\mathbf{v}}) \left(\mathbf{v}_f^{(n)} - \overline{\mathbf{v}_f^{(n)}} \right) \cdot \mathbf{S}_f + \rho_f \frac{\overline{a_f^{\circ} \mathbf{D}_f^{\mathbf{v}}}}{V_f} \left(\mathbf{v}_f^{\circ} - \overline{\mathbf{v}_f^{\circ}} \right) \cdot \mathbf{S}_f \end{aligned} \tag{E.3}$$

E.2 Anti-Diffusion Term

Other treatments of the anti-diffusion term in the volume fraction equations are stated here.

E.2.1 Additional Method 1

This method presented in the work of Rusche [20], suggests the use of maximum velocity magnitude in the transition region. The relative velocity at face is defined as:

$$\mathbf{v}_{r_f} = K_c \hat{\mathbf{n}}^* \max \left[\left\| \hat{\mathbf{n}}^* (\mathbf{v}_f \cdot \mathbf{n}) \right\| \right] \tag{E.4}$$

where K_c is a an adjustable factor, commonly set to 1.5. $\mathbf{n}$ is the unit normal of the face. $\hat{\mathbf{n}}^*$ is the unit normal of the interface of the curvature:

$$\hat{\mathbf{n}}^* = \frac{\nabla \alpha_f^{k,*}}{\|\nabla \alpha_f^{k,*}\| + \delta} \tag{E.5}$$

and $\alpha_f^{k,*}$ is the face interpolated value of $\alpha^{k,*}$, a smoothed volume fraction field, calculated using an elliptic relaxation equation:

$$\nabla \cdot \left[\left(\frac{c_\alpha}{\mathbf{d}} \right)^2 \nabla \alpha^{k,*} \right] = \alpha^{k,*} + \alpha^k \tag{E.6}$$

where c_α is an adjustable value, usually taken to be 0.5. $\mathbf{d}$ is a vector connecting the centers of elements straddling the face (when converted into surface integral as per Gauss theorem).

E.2.2 Additional Method 2

This method is addressed by [21]. It is proposed as an alternative to the previous methods which may give rise to a wrinkled interface. The current method provides a sharp yet smooth interface. $\mathbf{v}_{r_f}$ is defined such as:

$$\mathbf{v}_{r_f} = \Lambda_f \min \left[\zeta |\mathbf{v}_f \cdot \mathbf{n}|, \max_{\forall i} \left(|\mathbf{v}_{f_i} \cdot \mathbf{n}_i| \right) \right] \hat{\mathbf{n}} \tag{E.7}$$

where $\mathbf{v}_{r_f}$ is bounded to the maximum face velocity in the interface region. Λ_f is a weighting factor, defined as:

$$\Lambda_f = \min\left(\frac{\cos 2\theta_f + 1}{2}, 1\right), \quad \theta_f = \arccos|\hat{\mathbf{n}} \cdot \hat{\mathbf{n}}_f| \tag{E.8}$$

and ζ is fixed between 1 and 2. $\hat{\mathbf{n}}$ and $\mathbf{n}$ are respectively the curvature normal of the interface and the face unit normal. The curvature normal $\hat{\mathbf{n}}$ is calculated as:

$$\hat{\mathbf{n}} = \frac{\nabla\alpha_f^k}{\|\nabla\alpha_f^k\| + \delta} \tag{E.9}$$

and δ is a stabilizing factor, taken to be 10^{-5}, to avoid dividing by 0. α_f^k is the interpolated value of α^k to the face using a high resolution scheme, possibly the flux-limited second order QUICK scheme [131].

Appendix F

Extras

F.1 Special Types of Cases

F.1.1 Open Channel Cases

The critical conditions of the flow are determined by Froude's number Fr defined as:

$$Fr = \frac{V}{\sqrt{gy}}$$

where g is the gravitational acceleration magnitude, and y is a length scale usually taken as the free surface level above ground at inlet. if $Fr < 1$, then the flow is subcritical and disturbances can travel upstream as well as downstream, while if $Fr > 1$, then the flow is supercritical and disturbances cannot travel upstream. However, in both cases, assuming an inviscid flow, the free surface level will restore its height, unless the flow experiences a transition from subcritical to supercritical; this is dependent on the bump height Z_b and a critical bump height Z_{bc}, defined as:

$$Z_{bc} = \frac{y}{2}(Fr^{2/3} - 1)^2(Fr^{2/3} + 2)$$

if $Z_b = Z_{bc}$, then there's a transition to supercritical conditions, whereas if $Z_b > Z_{bc}$ then the flow preserves its conditions.

F.1.2 Closed Incompressible Cases

For cases that do not involve prescribed pressure boundary conditions (i.e. cavity), an additional treatment to the discrete system may be made in order to ensure a well-posed problem. An ill-posed problem can be reached in cases where only Neumann boundary conditions are imposed at the boundaries. For example in a closed system, where all the boundaries are walls, the pressure equation, which takes the form of a Poisson equation, is ill-posed because the boundary condition at the wall for the pressure field is of Neumann type, i.e. zero-gradient.

So there's a unique solution of the pressure equation up to a constant. Thus, an additional constraint has to be set in order to reach a unique solution. A very common treatment is to fix the pressure in the domain to some value at an arbitrary element. This could be done in different ways:

Direct Adjustment of Coefficient Matrix

One way to fix the pressure at an element is to alter the coefficients matrix such that the resultant pressure in that element reaches some value. This can be established by zeroing all the extra-diagonal entries of the matrix at the corresponding row, in order to eliminate their influence on the resultant value. And regarding the right-hand-side of the algebraic system, because the system is assembled in residual form, then the right-hand-side has to be also zeroed. Thus, considering the algebraic equation of the pressure equation in the coupled VOF system at an arbitrary element C^*:

$$
\begin{aligned}
& a^{pu}_{C^*} u_{C^*} + a^{pv}_{C^*} v_{C^*} + a^{pw}_{C^*} w_{C^*} + a^{pp}_{C^*} p_{C^*} + \sum_{k=1}^{n} a^{p\alpha^k}_{C^*} \alpha^k_{C^*} \\
& + \sum_{F\sim NB(C^*)} a^{pu}_F u_F + \sum_{F\sim NB(C^*)} a^{pv}_F v_F + \sum_{F\sim NB(C^*)} a^{pw}_F w_F + \sum_{F\sim NB(C^*)} a^{pp}_F p_F + \sum_{k=1}^{n} \sum_{F\sim NB(C^*)} a^{p\alpha^k}_F \alpha^k_F = b^p_{C^*}
\end{aligned}
$$

all the extra-diagonal coefficients as well as the source are set to zero:

$$
\begin{aligned}
a^{pu}_{C^*} &= 0 \quad a^{pv}_{C^*} = 0 \quad a^{pw}_{C^*} = 0 \\
a^{pu}_{F} &= 0 \quad a^{pv}_{F} = 0 \quad a^{pw}_{F} = 0 \\
b^{p}_{C^*} &= 0
\end{aligned}
$$

Pressure Shift to Satisfy Constraint

Another way to fix the pressure at the given element is by zeroing the pressure correction at that element, thus leaving the initial value of pressure at the element unchanged throughout. But in order to ensure consistency in the solution, and hence to ensure that mass conservation will not be violated, the value of the pressure correction at that element has also to be deducted from the whole pressure correction vector; this course of action is applicable because the current situation considers only a Poisson-like equation, which means that adjusting the level of pressure correction does not alter the solution. To demonstrate the concept, let $[p']$ be the resultant pressure correction, after solving the system, to the previous iteration pressure vector $[p^*]$. The pressure correction vector is shifted by a value equal to the pressure correction at an arbitrary element, say C^*. Then, $[p']$ is equal to:

$$
[p'] := [p'] - p'_{C^*} \qquad \text{(F.1)}
$$

In this way, the pressure correction at C^* would be 0, while the other pressure correction values at the other elements are shifted to maintain consistency. This consistency will ensure that mass conservation is not violated.

Bibliography

[1] J. Ha, P. Cleary, V. Alguine, and T. Nguyen, “Simulation of die filling in gravity die casting using sph and magmasoft,” in *Proc. 2nd Int. Conf. on CFD in Minerals & Process Industries*, pp. 423–428, 1999.

[2] P. W. Cleary and J. Ha, “Three dimensional modelling of high pressure die casting,” *International Journal of Cast Metals Research*, vol. 12, no. 6, pp. 357–365, 2000.

[3] M. Darwish and F. Moukalled, “Convective schemes for capturing interfaces of free-surface flows on unstructured grids,” *Numerical heat transfer, part B: Fundamentals*, vol. 49, no. 1, pp. 19–42, 2006.

[4] F. Moukalled and M. Darwish, “Transient schemes for capturing interfaces of free-surface flows,” *Numerical Heat Transfer, Part B: Fundamentals*, vol. 61, no. 3, pp. 171–203, 2012.

[5] A. Faghri and Y. Zhang, *Transport phenomena in multiphase systems.* Elsevier, 2006.

[6] F. H. Harlow and J. E. Welch, “Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface,” *The physics of fluids*, vol. 8, no. 12, pp. 2182–2189, 1965.

[7] B. Nichols and C. Hirt, “Methods for calculating multidimensional, transient free surface flows past bodies,” in *Proc., 1st Int. Conf. Ship Hydrodynamics*, pp. 253–277, Naval Ship Research and Development Center, Bethesda, Md, 1975.

[8] C. W. Hirt and B. D. Nichols, “Volume of fluid (vof) method for the dynamics of free boundaries,” *Journal of computational physics*, vol. 39, no. 1, pp. 201–225, 1981.

[9] O. Ubbink, “Numerical prediction of two fluid systems with sharp interfaces,” 1997.

[10] V. Efremov, A. S. Kozelkov, A. Kornev, A. Kurkin, V. V. Kurulin, D. Y. Strelets, and N. V. Tarasova, “Method for taking into account gravity in free-surface flow simulation,” *Computational Mathematics and Mathematical Physics*, vol. 57, no. 10, pp. 1720–1733, 2017.

[11] P. J. Zwart, P. G. Godin, J. Penrose, and S. H. Rhee, “Simulation of unsteady free-surface flow around a ship hull using a fully coupled multi-phase flow method,” *Journal of marine science and technology*, vol. 13, no. 4, p. 346, 2008.

[12] V. K. Gupta, K. Srikanth, and H. Punekar, “Improvements in free surface flow numerics using coupled vof and pseudo transient solver,” in *2016 IEEE 23rd International Conference on High Performance Computing Workshops (HiPCW)*, pp. 100–105, IEEE, 2016.

[13] B. Leonard and S. Mokhtari, “Beyond first-order upwinding: The ultra-sharp alternative for non-oscillatory steady-state simulation of convection,” *International Journal for Numerical Methods in Engineering*, vol. 30, no. 4, pp. 729–766, 1990.

[14] O. Ubbink and R. Issa, “A method for capturing sharp fluid interfaces on arbitrary meshes,” *Journal of Computational Physics*, vol. 153, no. 1, pp. 26–50, 1999.

[15] S. Muzaferija, “Computation of free surface flows using interface-tracking and interface-capturing methods,” *Nonlinear water-wave interaction. Computational Mechanics, Southampton*, 1998.

[16] M. Hoekstra, G. Vaz, B. Abeil, and T. Bunnik, “Free-surface flow modelling with interface capturing techniques,” *MARINE2007*, vol. 2, pp. 1–4, 2007.

[17] K. So, X. Hu, and N. A. Adams, “Anti-diffusion method for interface steepening in two-phase incompressible flow,” *Journal of Computational Physics*, vol. 230, no. 13, pp. 5155–5177, 2011.

[18] K. So, X. Hu, and N. A. Adams, “Anti-diffusion interface sharpening technique for two-phase compressible flow simulations,” *Journal of Computational Physics*, vol. 231, no. 11, pp. 4304–4323, 2012.

[19] J. Klostermann, K. Schaake, and R. Schwarze, “Numerical simulation of a single rising bubble by vof with surface compression,” *International Journal for Numerical Methods in Fluids*, vol. 71, no. 8, pp. 960–982, 2013.

[20] H. Rusche, *Computational fluid dynamics of dispersed two-phase flows at high phase fractions.* PhD thesis, University of London, 2002.

[21] Y. Mehmani, “Wrinkle-free interface compression for two-fluid flows,” *arXiv preprint arXiv:1811.09744*, 2018.

[22] T. Barth and D. Jespersen, “The design and application of upwind schemes on unstructured meshes,” in *27th Aerospace sciences meeting*, p. 366, 1989.

[23] E. Shirani, A. Jafari, and N. Ashgriz, “Turbulence models for flows with free surfaces and interfaces,” *AIAA journal*, vol. 44, no. 7, pp. 1454–1462, 2006.

[24] A. Balabel, “A generalized level set-navier stokes numerical method for predicting thermo-fluid dynamics of turbulent free surface,” *Computer Modeling in Engineering & Sciences(CMES)*, vol. 83, no. 6, pp. 599–638, 2012.

[25] D. Fuster, G. Agbaglah, C. Josserand, S. Popinet, and S. Zaleski, “Numerical simulation of droplets, bubbles and waves: state of the art,” *Fluid dynamics research*, vol. 41, no. 6, p. 065001, 2009.

[26] B. Lafaurie, C. Nardone, R. Scardovelli, S. Zaleski, and G. Zanetti, “Modelling merging and fragmentation in multiphase flows with surfer,” *Journal of Computational Physics*, vol. 113, no. 1, pp. 134–147, 1994.

[27] J. R. Taylor, S. Sarkar, and V. Armenio, “Large eddy simulation of stably stratified open channel flow,” *Physics of Fluids*, vol. 17, no. 11, p. 116602, 2005.

[28] W.-L. Hong and D. T. Walker, “Reynolds-averaged equations for free-surface flows with application to high-froude-number jet spreading,” *Journal of Fluid Mechanics*, vol. 417, pp. 183–209, 2000.

[29] V. Ferreira, N. Mangiavacchi, M. Tomé, A. Castelo, J. Cuminato, and S. McKee, “Numerical simulation of turbulent free surface flow with two-equation k–ε eddy-viscosity models,” *International Journal for Numerical Methods in Fluids*, vol. 44, no. 4, pp. 347–375, 2004.

[30] S. B. Pope and S. B. Pope, *Turbulent flows.* Cambridge university press, 2000.

[31] S. Pope, “A more general effective-viscosity hypothesis,” *Journal of Fluid Mechanics*, vol. 72, no. 2, pp. 331–340, 1975.

[32] T. B. Gatski and C. G. Speziale, “On explicit algebraic stress models for complex turbulent flows,” *Journal of fluid Mechanics*, vol. 254, pp. 59–78, 1993.

[33] S. Wallin and A. V. Johansson, “An explicit algebraic reynolds stress model for incompressible and compressible turbulent flows,” *Journal of Fluid Mechanics*, vol. 403, pp. 89–132, 2000.

[34] D. Violeau and R. Issa, “Numerical modelling of complex turbulent free-surface flows with the sph method: an overview,” *International Journal for Numerical Methods in Fluids*, vol. 53, no. 2, pp. 277–304, 2007.

[35] R. L. Panton, *Incompressible flow.* John Wiley & Sons, 2013.

[36] L. Landau and E. Lifshitz, “Fluid mechanics 2nd edition pergamon press,” 1987.

[37] P. K. Kundu and I. Cohen, “Fluid mechanics, academic press,” *Philadelphia, Pennsylvania*, 1990.

[38] C. T. Crowe, *Multiphase flow handbook*, vol. 59. CRC press, 2005.

[39] N. D. Katopodes, *Free-surface Flow: Environmental Fluid Mechanics.* Butterworth-Heinemann, 2018.

[40] T. B. Anderson and R. Jackson, “Fluid mechanical description of fluidized beds. equations of motion,” *Industrial & Engineering Chemistry Fundamentals*, vol. 6, no. 4, pp. 527–539, 1967.

[41] R. M. Bowen, “Theory of mixtures in continuum physics,” *Mixtures and Electromagnetic Field Theories*, vol. 3, 1976.

[42] D. Zhilenko, O. Krivonosova, M. Gritsevich, and P. Read, “Wave number selection in the presence of noise: Experimental results,” *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 28, no. 5, p. 053110, 2018.

[43] J. P. Heller, “An unmixing demonstration,” *American Journal of Physics*, vol. 28, no. 4, pp. 348–353, 1960.

[44] D. Kuzmin and J. Hämäläinen, *Methods for Computational Fluid Dynamics: A Practical Guide.* SIAM, 2014.

[45] J. D. Anderson and J. Wendt, *Computational fluid dynamics*, vol. 206. Springer, 1995.

[46] T. Chung, *Computational fluid dynamics.* Cambridge university press, 2010.

[47] R. J. LeVeque *et al.*, *Finite volume methods for hyperbolic problems*, vol. 31. Cambridge university press, 2002.

[48] S. V. Patankar and D. B. Spalding, “A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows,” in *Numerical prediction of flow, heat transfer, turbulence and combustion*, pp. 54–73, Elsevier, 1983.

[49] M. Darwish, F Moukalled, “A unified formulation of the segregated class of algorithms for fluid flow at all speeds,” *Numerical Heat Transfer: Part B: Fundamentals*, vol. 37, no. 1, pp. 103–139, 2000.

[50] I. Demirdžić, Ž. Lilek, and M. Perić, “A collocated finite volume method for predicting flows at all speeds,” *International Journal for Numerical Methods in Fluids*, vol. 16, no. 12, pp. 1029–1050, 1993.

[51] S. Godunov, “A difference scheme for numerical computation of discontinuous solutions of fluid dynamics,” *Mat. Sb*, vol. 47, pp. 271–306, 1959.

[52] P. Lax and B. Wendroff, “Systems of conservation laws. communications on pure and applied mathematics,” 1960.

[53] C. Hung and R. MacCormack, “Numerical solutions of supersonic and hypersonic laminar compression corner flows,” *AIAA Journal*, vol. 14, no. 4, pp. 475–481, 1976.

[54] R. W. MacCormack, “A numerical method for solving the equations of compressible viscous flow,” *AIAA journal*, vol. 20, no. 9, pp. 1275–1281, 1982.

[55] Y.-H. Choi and C. L. Merkle, “The application of preconditioning in viscous flows,” *Journal of computational physics*, vol. 105, no. 2, pp. 207–223, 1993.

[56] C. Hirsch, “Numerical computation of internal and external flows, volume 1: Fundamentals of numerical discretization,” *John Wiley and Sons*, vol. 9, p. 10, 1988.

[57] F. Moukalled, L. Mangani, M. Darwish, *et al.*, “The finite volume method in computational fluid dynamics,” *An advanced introduction with OpenFoam® and Matlab®. Nueva York: Springer. Recuperado de http://www. gidropraktikum. narod. ru/Moukalled-et-al-FVM-OpenFOAM-Matlab. pdf*, 2016.

[58] S. V. Patankar, “Numerical heat transfer and fluid flow(book),” *Washington, DC, Hemisphere Publishing Corp., 1980. 210 p*, 1980.

[59] M. Darwish, D. Asmar, and F. Moukalled, “A comparative assessment within a multigrid environment of segregated pressure-based algorithms for fluid flow at all speeds,” *Numerical Heat Transfer, Part B: Fundamentals*, vol. 45, no. 1, pp. 49–74, 2004.

[60] J. Van Doormaal and G. Raithby, “An evaluation of the segregated approach for predicting incompressible fluid flows,” *ASME paper*, no. 85-HT, p. 9, 1985.

[61] M. Darwish, A. Abdel Aziz, and F. Moukalled, “A coupled pressure-based finite-volume solver for incompressible two-phase flow,” *Numerical Heat Transfer, Part B: Fundamentals*, vol. 67, no. 1, pp. 47–74, 2015.

[62] L. Mangani, M. Buchmayr, and M. Darwish, “Development of a novel fully coupled solver in openfoam: Steady-state incompressible turbulent flows,” *Numerical Heat Transfer, Part B: Fundamentals*, vol. 66, no. 1, pp. 1–20, 2014.

[63] M. Darwish, I. Sraj, and F. Moukalled, “A coupled finite volume solver for the solution of incompressible flows on unstructured grids,” *Journal of Computational Physics*, vol. 228, no. 1, pp. 180–201, 2009.

[64] L. Mangani, M. Darwish, and F. Moukalled, “An openfoam pressure-based coupled cfd solver for turbulent and compressible flows in turbomachinery applications,” *Numerical Heat Transfer, Part B: Fundamentals*, vol. 69, no. 5, pp. 413–431, 2016.

[65] C. Fernandes, V. Vukčević, T. Uroić, R. Simoes, O. Carneiro, H. Jasak, and J. Nóbrega, “A coupled finite volume flow solver for the solution of incompressible viscoelastic flows,” *Journal of Non-Newtonian Fluid Mechanics*, vol. 265, pp. 99–115, 2019.

[66] C.-N. Xiao, F. Denner, and B. G. van Wachem, “Fully-coupled pressure-based finite-volume framework for the simulation of fluid flows at all speeds in complex geometries,” *Journal of Computational Physics*, vol. 346, pp. 91–130, 2017.

[67] T. Uroić and H. Jasak, “Block-selective algebraic multigrid for implicitly coupled pressure-velocity system,” *Computers & fluids*, vol. 167, pp. 100–110, 2018.

[68] F. Denner, “Fully-coupled pressure-based algorithm for compressible flows: Linearisation and iterative solution strategies,” *Computers & Fluids*, vol. 175, pp. 53–65, 2018.

[69] G. G. Ferreira, P. L. Lage, L. F. L. Silva, and H. Jasak, “Implementation of an implicit pressure–velocity coupling for the eulerian multi-fluid model,” *Computers & Fluids*, vol. 181, pp. 188–207, 2019.

[70] M. Riella, R. Kahraman, and G. Tabor, “Fully-coupled pressure-based two-fluid solver for the solution of turbulent fluid-particle systems,” *Computers & Fluids*, vol. 192, p. 104275, 2019.

[71] F. Denner and B. G. van Wachem, “Numerical time-step restrictions as a result of capillary waves,” *Journal of Computational Physics*, vol. 285, pp. 24–40, 2015.

[72] K. Kissling, J. Springer, H. Jasak, S. Schutz, K. Urban, and M. Piesche, “A coupled pressure based solution algorithm based on the volume-of-fluid approach for two or more immiscible fluids,” in *Proceedings of the V European Conference on Computational Fluid Dynamics ECCOMAS CFD 2010*, 2010.

[73] F. Moukalled and M. Darwish, “On the performance of mass conservation based algorithms for multi-phase flows,”

[74] F. Moukalled and M. Darwish, “A unified formulation of the segregated class of algorithms for multi-fluid flow at all-speeds,” 1999.

[75] J. U. Brackbill, D. B. Kothe, and C. Zemach, “A continuum method for modeling surface tension,” *Journal of computational physics*, vol. 100, no. 2, pp. 335–354, 1992.

[76] E. Berberović, N. P. van Hinsberg, S. Jakirlić, I. V. Roisman, and C. Tropea, “Drop impact onto a liquid layer of finite thickness: Dynamics of the cavity evolution,” *Physical Review E*, vol. 79, no. 3, p. 036306, 2009.

[77] B. Leonard, “The ultimate conservative difference scheme applied to unsteady one-dimensional advection,” *Computer methods in applied mechanics and engineering*, vol. 88, no. 1, pp. 17–74, 1991.

[78] C. M. Rhie and W. L. Chow, “Numerical study of the turbulent flow past an airfoil with trailing edge separation,” *AIAA Journal*, vol. 21, no. 11, pp. 1525–1532, 1983.

[79] L. Landau, “Em lifshitz, fluid mechanics,” *Course of theoretical physics*, vol. 6, 1959.

[80] J. H. Ferziger and M. Peric, *Computational methods for fluid dynamics.* Springer Science & Business Media, 2012.

[81] H. Jasak, “Error analysis and estimation for the finite volume method with applications to fluid flows. thesis submitted for the degree of doctor. department of mechanical engineering, imperial college of science, 1996.,” 1996.

[82] C. A. Fletcher, *Computational techniques for fluid dynamics 2: Specific techniques for different flow categories.* Springer Science & Business Media, 2012.

[83] C. Gu, “Computation of flows with large body forces,” *Numerical methods in laminar and turbulent flow*, vol. 7, no. Part 2, pp. 1568–1578, 1991.

[84] J. Mencinger, “An alternative finite volume discretization of body force field on collocated grid,” in *Finite Volume Method-Powerful Means of Engineering Design*, InTech, 2012.

[85] D. Z. Zhang, W. B. VanderHeyden, Q. Zou, X. Ma, and P. T. Giguere, “Cartablanca theory manual: Multiphase flow equations and numerical methods,” in *Technical Report LAUR-07-3621*, pp. 07–3621, Los Alamos National Lab NM, 2007.

[86] A. Khrabry, E. Smirnov, D. Zaytsev, and V. Goryachev, “Numerical study of 2d and 3d separation phenomena in the dam-break flow interacting with a triangular obstacle,” *Periodica Polytechnica Mechanical Engineering*, vol. 60, no. 3, pp. 159–166, 2016.

[87] H. Joshi, “Balanced-force numerical method for two phase flow at the onset of instability,” 2017.

[88] S. M. Damián and N. M. Nigro, “An extended mixture model for the simultaneous treatment of small-scale and large-scale interfaces,” *International Journal for Numerical Methods in Fluids*, vol. 75, no. 8, pp. 547–574, 2014.

[89] M. Darwish, “Development and testing of a robust free-surface finite volume method,” *Faculty of Engineering and Architecture, American University of Beirut*, 2003.

[90] 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*, vol. 221, no. 2, pp. 524–538, 2007.

[91] F. Denner and B. G. van Wachem, “Fully-coupled balanced-force vof framework for arbitrary meshes with least-squares curvature evaluation from volume fractions,” *Numerical Heat Transfer, Part B: Fundamentals*, vol. 65, no. 3, pp. 218–255, 2014.

[92] 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*, vol. 43, no. 3, pp. 267–287, 2003.

[93] J. H. Ferziger, “Interfacial transfer in tryggvason’s method,” *International Journal for Numerical Methods in Fluids*, vol. 41, no. 5, pp. 551–560, 2003.

[94] D.-J. Yum and B.-S. Yoon, “Numerical simulation of slamming phenomena for 2-d wedges,” *Journal of the Society of Naval Architects of Korea*, vol. 45, no. 5, pp. 477–486, 2008.

[95] B.-H. Lee, S.-J. Jung, M.-C. Ryu, Y.-S. Kim, and J.-C. Park, "Numerical simulation for fluid impact loads by flat plate with incident angles," *Journal of the Society of Naval Architects of Korea*, vol. 45, no. 1, pp. 1–9, 2008.

[96] H. P. Tran and H.-T. Ahn, "Air compressibility effect in cfd-based water impact analysis," *Journal of the Society of Naval Architects of Korea*, vol. 48, no. 6, pp. 581–591, 2011.

[97] J. A. Heyns, A. G. Malan, T. M. Harms, and O. F. Oxtoby, "A weakly compressible free-surface flow solver for liquid–gas systems using the volume-of-fluid approach," *Journal of Computational Physics*, vol. 240, pp. 145–157, 2013.

[98] G. Raithby, P. Galpin, and J. Van Doormaal, "Prediction of heat and fluid flow in complex geometries using general orthogonal coordinates," *Numerical Heat Transfer, Part A: Applications*, vol. 9, no. 2, pp. 125–142, 1986.

[99] J. Liu, C. R. Kaplan, and E. S. Oran, "A brief note on implementing boundary conditions at a solid wall using the fct algorithm," tech. rep., NAVAL RESEARCH LAB WASHINGTON DC, 2006.

[100] C. Pommerell and W. Fichtner, "Memory aspects and performance of iterative solvers," *SIAM Journal on Scientific Computing*, vol. 15, no. 2, pp. 460–473, 1994.

[101] B. Hutchinson and G. Raithby, "A multigrid method based on the additive correction strategy," *Numerical Heat Transfer, Part A: Applications*, vol. 9, no. 5, pp. 511–537, 1986.

[102] K. Stüben, "An introduction to algebraic multigrid. inmultigrid, ed. u. trottenberg, cw ooosterlee, and a. schüller, appendix, 413-532," 2001.

[103] R. P. Fedorenko, "A relaxation method for solving elliptic difference equations," *USSR Computational Mathematics and Mathematical Physics*, vol. 1, no. 4, pp. 1092–1096, 1962.

[104] F. De la Vallee Poussin and W. P. Timlake, "An accelerated relaxation algorithm for iterative solution of elliptic equations," *SIAM Journal on Numerical Analysis*, vol. 5, no. 2, pp. 340–351, 1968.

[105] A. Brandt, "Multi-level adaptive solutions to boundary-value problems," *Mathematics of computation*, vol. 31, no. 138, pp. 333–390, 1977.

[106] V. R. Gopala and B. G. van Wachem, "Volume of fluid methods for immiscible-fluid and free-surface flows," *Chemical Engineering Journal*, vol. 141, no. 1-3, pp. 204–221, 2008.

[107] S. T. Zalesak, “Fully multidimensional flux-corrected transport algorithms for fluids,” *Journal of computational physics*, vol. 31, no. 3, pp. 335–362, 1979.

[108] J. B. Bell, P. Colella, and H. M. Glaz, “A second-order projection method for the incompressible navier-stokes equations,” *Journal of Computational Physics*, vol. 85, no. 2, pp. 257–283, 1989.

[109] W. J. Rider and D. B. Kothe, “Reconstructing volume tracking,” *Journal of computational physics*, vol. 141, no. 2, pp. 112–152, 1998.

[110] D. Reed, J. Yu, H. Yeh, and S. Gardarsson, “Investigation of tuned liquid dampers under large amplitude excitation,” *Journal of engineering mechanics*, vol. 124, no. 4, pp. 405–413, 1998.

[111] K. Kleefsman, G. Fekken, A. Veldman, B. Iwanowski, and B. Buchner, “A volume-of-fluid based simulation method for wave impact problems,” *Journal of computational physics*, vol. 206, no. 1, pp. 363–393, 2005.

[112] R. ISSA, D. VIOLEAU, and E. De France, “3d dambreaking,” *SPH European Research Interest Community SIG*, 2006.

[113] I. Campbell and P. Weynberg, *Measurement of parameters affecting slamming*. University of Southampton, Department of Aeronautics and Astronautics, 1980.

[114] B. Koo, J. Yang, S. M. Yeon, and F. Stern, “Reynolds and froude number effect on the flow past an interface-piercing circular cylinder,” *International Journal of Naval Architecture and Ocean Engineering*, vol. 6, no. 3, pp. 529–561, 2014.

[115] M. Inoue, N. Baba, and Y. Himeno, “Experimental and numerical study of viscous flow field around an advancing vertical circular cylinder piercing a free-surface,” 1993.

[116] T. Kawamura, S. Mayer, A. Garapon, and L. Sørensen, “Large eddy simulation of a flow past a free surface piercing circular cylinder,” *J. Fluids Eng.*, vol. 124, no. 1, pp. 91–101, 2002.

[117] G. Yu, E. Avital, and J. Williams, “Large eddy simulation of flow past free surface piercing circular cylinders,” *Journal of Fluids Engineering*, vol. 130, no. 10, 2008.

[118] J. Suh, J. Yang, and F. Stern, “The effect of air–water interface on the vortex shedding from a vertical circular cylinder,” *Journal of Fluids and Structures*, vol. 27, no. 1, pp. 1–22, 2011.

[119] J. R. Chaplin and P. Teigen, “Steady flow past a vertical surface-piercing circular cylinder,” *Journal of Fluids and Structures*, vol. 18, no. 3-4, pp. 271–285, 2003.

[120] K. J. Bai, J. H. Han, *et al.*, “A localized finite-element method for the nonlinear steady waves due to a two-dimensional hydrofoil,” *Journal of Ship Research*, vol. 38, no. 01, pp. 42–51, 1994.

[121] S. Bal, “Prediction of wave pattern and wave resistance of surface piercing bodies by a boundary element method,” *International Journal for Numerical Methods in Fluids*, vol. 56, no. 3, pp. 305–329, 2008.

[122] H. Ghassemi and A. R. Kohansal, “Isoparametric boundary element method applied to the rectangular and delta hydrofoils near the free surface,” *China ocean engineering*, vol. 24, no. 2, pp. 321–331, 2010.

[123] J. Kouh, “Performance analysis of two-dimensional hydrofoil under free surface,” *J Eng Natl Taiwan Univ*, vol. 86, pp. 113–123, 2002.

[124] J. H. Duncan, “The breaking and non-breaking wave resistance of a two-dimensional hydrofoil,” *Journal of Fluid Mechanics*, vol. 126, pp. 507–520, 1983.

[125] J. Blazek, *Computational fluid dynamics: principles and applications*. Butterworth-Heinemann, 2015.

[126] T.-H. Shih, W. W. Liou, A. Shabbir, Z. Yang, and J. Zhu, “A new k-epsilon eddy viscosity model for high reynolds number turbulent flows,” *Computers & fluids*, vol. 24, no. 3, pp. 227–238, 1995.

[127] F. R. Menter, “Two-equation eddy-viscosity turbulence models for engineering applications,” *AIAA journal*, vol. 32, no. 8, pp. 1598–1605, 1994.

[128] S. Spekreijse, “Multigrid solution of monotone second-order discretizations of hyperbolic conservation laws,” *Mathematics of Computation*, vol. 49, no. 179, pp. 135–155, 1987.

[129] G.-H. Kim and S. Park, “Development of a numerical simulation tool for efficient and robust prediction of ship resistance,” *International Journal of Naval Architecture and Ocean Engineering*, vol. 9, no. 5, pp. 537–551, 2017.

[130] S.-E. Kim, “A multi-dimensional linear reconstruction scheme for arbitrary unstructured mesh,” in *16th AIAA Computational Fluid Dynamics Conference*, p. 3990, 2003.

[131] B. P. Leonard, “Simple high-accuracy resolution program for convective modelling of discontinuities,” *International journal for numerical methods in fluids*, vol. 8, no. 10, pp. 1291–1318, 1988.