Computational Methods for Simulations of Multiphase Compressible Flows for Atomization Applications

by

Karthik Kannan

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved April 2020 by the
Graduate Supervisory Committee:

Marcus Herrmann, Chair
Huei-Ping Huang
Juan Lopez
Yulia Peet
Liping Wang

ARIZONA STATE UNIVERSITY

May 2020

ABSTRACT

Compressible fluid flows involving multiple physical states of matter occur in both nature and technical applications such as underwater explosions and implosions, cavitation-induced bubble collapse in naval applications and Richtmyer-Meshkov type instabilities in inertial confinement fusion. Of particular interest is the atomization of fuels that enable shock-induced mixing of fuel and oxidizer in supersonic combustors. Due to low residence times and varying length scales, providing insight through physical experiments is both technically challenging and sometimes unfeasible. Numerical simulations can help provide detailed insight and aid in the engineering design of devices that can harness these physical phenomena.

In this research, computational methods were developed to accurately simulate phase interfaces in compressible fluid flows with a focus on targeting primary atomization. Novel numerical methods which treat the phase interface as a discontinuity, and as a smeared region were developed using low-dissipation, high-order schemes. The resulting methods account for the effects of compressibility, surface tension and viscosity. To aid with the varying length scales and high-resolution requirements found in atomization applications, an adaptive mesh refinement (AMR) framework is used to provide high-resolution only in regions of interest. The developed methods were verified with test cases involving strong shocks, high density ratios, surface tension effects and jumps in the equations of state, in one-, two- and three-dimensions, obtaining good agreement with theoretical and experimental results. An application case of the primary atomization of a liquid jet injected into a Mach 2 supersonic crossflow of air is performed with the methods developed.

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to Prof. Marcus Herrmann for providing immense support and guidance throughout the course of this research work. I would also like to thank my colleagues at Arizona State University and all the conversations along the years which has helped in many ways. I would also like to express my gratitude towards Taitech Inc. and the Airforce Research Laboratory under subcontract TS15-16-02-005 for providing funding support for this work. Finally, the computing resources provided by Research Computing at Arizona State University contributed towards making the results in this research possible and is acknowledged.

TABLE OF CONTENTS

CHAPTER Page

LIST OF TABLES

Table Page

Table Page

LIST OF FIGURES

Figure Page

Figure Page

Figure Page

Figure Page

Chapter 1

INTRODUCTION

In this chapter, the background and motivation for the proposed research is described. Specifically, this chapter will detail the physical applications of the current work, the problem to be solved and provide an overview of the existing approaches to the problem in literature.

1.1 Atomization in High Speed Flows

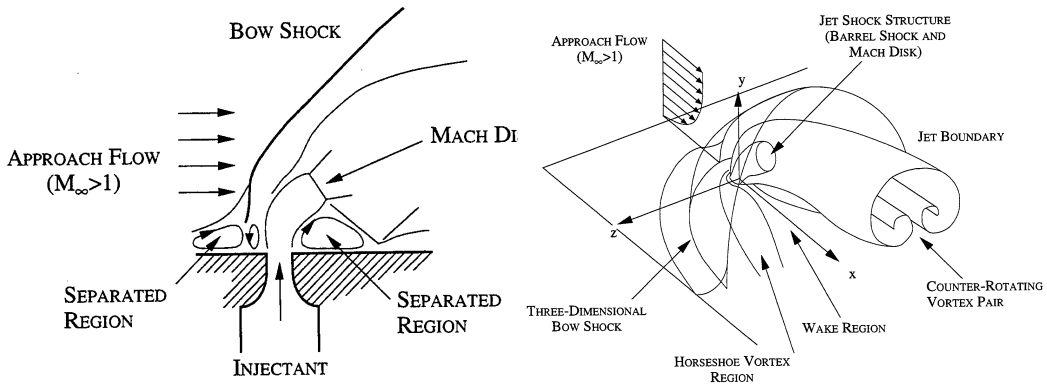
Atomization is the process of breaking down bulk volumes of liquid into smaller structures, and can be observed fairly common in nature. In addition to scientific motivation, the importance of atomization in a wide variety of applications proves to be of technical interest. Examples include fuel injection in combustion engines, pharmaceutical applications, and food processing applications. Atomization can also take place in high speed flows due to turbulence, shear forces and more importantly the interaction with shocks. Examples of atomization applications in high speed flows include Richtmyer–Meshkov type instabilities in confinement fusion and supernova explosions, cavitation processes in naval applications, high speed fuel injection in modern diesel and gasoline engines. Another application of modern interest is the study of combustion process in a scramjet (supersonic combustion ramjet) engine, which could prove to be one of the most effective engine cycles for hypersonic flight

[30]. Owing to a low fluid residence time of the order of milliseconds inside the combustor of the scramjet engine [66], both rapid mixing and combustion processes take place nearly simultaneously [68]. Therefore, understanding the mechanisms by which fuel is atomized and mixed with the surrounding air is crucial to the development of scramjet engines. Other early studies of atomization in high-speed flows were mostly geared towards applications to thrust vectoring [106].

1.2 Interaction of Jets in a Cross-Flow

The problem of a jet of fluid in a cross-flow of another fluid (often called jet-in-crossflow) has been studied widely because it models the physical system found in many of the aforementioned applications. In a typical jet-in-crossflow problem, a jet of fluid exits an orifice and interacts with another fluid which is flowing across the face of the orifice. The problem of liquid jets in subsonic crossflow has been studied fairly thoroughly with early experiments by Pratte and Baines [132] measuring jet trajectories for different flow conditions. More recent examples of experimental studies include [159, 146]. Significant computational efforts made using Large-Eddy Simulation (LES) or Direct Numerical Simulation (DNS) showcase the value of having full three-dimensional data to understand the underlying flow physics. Selected examples of computational efforts include [111, 59, 112]. The comprehensive outcome

of both experimental and numerical studies alike is the understanding of key flow physics such as the presence of downstream counter-rotating vortex pairs, horseshoe and wake vortices, and empirical relations for the jet trajectory.



(a) Cross-sectional schematic **(b)** Full 3D schematic

Figure 1: Schematic of a liquid jet in a supersonic crossflow (taken from [54])

On the other hand, the problem of liquid jets in supersonic crossflow is a relatively underdeveloped area, primarily due to its challenging nature both experimentally and computationally. Early studies using a gas dynamical perspective produced a fairly detailed qualitative description of time-averaged flow in the problem. Figure 1 shows the schematic of the qualitative flow field in a typical jet in supersonic crossflow due to experiments by Gruber, Nejad, and Dutton [54]. It can be observed that a three-dimensional bow shock sits upstream of the injected liquid jet and interacts with the boundary layer to create separation. The bow shock sweeps around the injector orifice and interacts with the separation shock. Moreover, a horseshoe vortex region

forms around the injector orifice as seen in Fig. 1b. Furthermore, a Mach disk is formed due to the acceleration of the jet core flow. Finally, a pair of counter-rotating vortices can be seen directly downstream of the injector orifice.

Quantitative data however, is less common than those at subsonic speeds [106] especially if turbulence or mixing data are to be measured. With increase in sophistication of experimental techniques, there have been efforts to better quantify the underlying problem. For example, using schlieren imaging techniques, Papamoschou and Hubbard [121] and Ben-Yakar, Mungal, and Hanson [10] have measured jet penetration into the supersonic flow. Detailed streamwise and crossplane particle image velocimetry (PIV) data for supersonic jets into subsonic crossflow have been provided due to experiments by Beresh et al. [11]. Quantitative data of concentrations for sonic ethylene injected into a Mach 2 air crossflow was provided by Lin et al. [97] using Raman scattering technique. The experimental efforts provide an insight into the overall structure of the jets and its trajectory, but is lacking quantitative data relative to the subsonic case.

On the computational side, most early efforts made use of algebraic mixing-length and eddy-viscosity models [54]. It was noted by Chenault, Beran, and Bowersox [22] that such models can only yield estimates of the overall flow structure but are incapable of correctly modeling secondary flow features. There have been several simulations of under-expanded sonic jets, for example, [79, 19]. Using hybrid Reynolds-Averaged Navier-Stokes (RANS)/LES approaches, simulations by Boles et al. [15] and Peterson and Candler [124] were performed to obtain good agreement with the aforementioned

experimental work by [97]. The outcome of the experimental and numerical efforts is a fairly detailed understanding of the qualitative time-averaged behavior. Due to increasing computational capabilities there also exists a fairly good understanding of the qualitative instantaneous behavior. On the quantitative side, several correlations for the trajectories of the high-speed jets have been proposed [54] based on experiments. Overall, based on the detailed insight provided by computational modeling in the subsonic case, it is very likely that the supersonic case will also benefit similarly.

1.3 Numerical Modeling of Compressible Multiphase Flows

The approach taken to modeling atomization is a hybrid approach where the entire process is split into two — primary and secondary. Primary atomization represents the initial breakup of the jet resulting in complex large coherent structures. The structures formed during primary atomization can undergo further breakup in a process known as secondary atomization, resulting in many smaller droplets. Under the hybrid approach, the physics of primary atomization are resolved and the phase interface is captured directly. Droplets below a certain threshold are transferred to a secondary atomization spray model, for example, the Lagrangian particle model. The present research will focus entirely on the primary atomization of compressible multiphase flows. A summary of the numerical methods developed in literature to this end is discussed below.

The various numerical methods can be classified based on how the phase interface is treated. The treatment can be categorized as either a Sharp Interface Method (SIM) or a Diffused Interface Method (DIM). Both these approaches have their advantages and disadvantages. Methods categorized as SIM are those that have its fluid properties change across the phase interface in a discontinuous manner, which mimics the physical nature of a phase interface. As a result, SIM approaches involve complex geometric considerations or special treatment at interfaces. These methods require some means of locating the fluid interface in the computational domain and then avoid dissipative errors across the interface by either solving the two-fluid Riemann problem at the fluid interface directly [25, 38] or by introducing so-called ghost states eliminating numerical stencils across the fluid interface discontinuity. The idea of ghost states was first introduced by Glimm, Marchesin, and McBryan [48] and later extended by Fedkiw et al. [42]. Ghost fluid methods use extrapolation to create stencil values across an interface, as having a computational stencil across an interface can produce inconsistent mass and momentum transfer for high density ratios. Selected examples of efforts using ghost fluid method include [100, 101, 63, 62, 161]. The formulation of Glimm, Marchesin, and McBryan [48] and Glimm et al. [51] on the other hand uses front tracking and maintains two states for cells containing the fluid interface, however, this may generate arbitrarily small sized cut cells that requires a merging algorithm of the cut cell with one of its neighbors. Arbitrary Lagrangian Eulerian (ALE) methods use a mesh which conforms to the phase interface

and deforms with the interface transport. As a result, it involves computationally expensive meshing and re-meshing techniques. Examples of ALE methods simulating compressible multiphase flows include [40, 104, 168, 31].

On the other hand, DIM approaches smear out the interface over a number of computational cells either due to dissipative numerical errors, or by design, by imposing a smooth transition function between the fluids. In the latter case, non-physical mixed states between the pure fluid states develop that may result in non-physical pressure oscillations in the solution [1, 2]. To avoid these, the captured interface motion must be described by the solution of advection equations in non-conservative form for specific combinations of material properties of the fluids [1, 2, 133, 153, 151, 74, 108]. DIM is straightforward to implement in higher dimensions and is analogous to the idea behind shock capturing in compressible flows. These methods are able to handle large topology changes in the interface with relative ease. To counter the numerical diffusion of the interface, these methods often employ an interface re-sharpening scheme. Examples include a post time-step anti-diffusion correction by So, Hu, and Adams [158], interface compression method introduced by Shukla, Pantano, and Freund [150] and later extended by Tiwari, Freund, and Pantano [164]. Finally, a conservative reconstruction based interface sharpening approach is the Tangent of Hyperbola for Interface Capturing (THINC) method [174], which unlike the previously mentioned sharpening approaches does not modify the

governing equations. DIM based approaches to simulating compressible two-phase flows are fairly mature due to their relatively simplistic nature, see [139] for a recent overview.

It is important to note that using either approach, many efforts neglect viscous effects and effects due to surface tension forces, and focus on the inviscid limit, for example [152, 81]. However, viscous forces and surface tension forces play an important role in atomization, as they dictate the temporal evolution of droplets in a flow field. Finally, of note is the effort by Xiao et al. [173] using LES to simulate primary breakup of liquid jet in supersonic crossflow. The method uses an incompressible solver to track the liquid phase and compressible solver to track the gas phase, coupling the two phases using suitable boundary conditions at the interface.

1.4 Outline of Research

As described earlier in this chapter, the main motivation behind the present research is to accurately simulate the primary atomization of a liquid in compressible flows. Based on the literature review concerning the state-of-the-art for modeling compressible multiphase flows and its shortcomings, the present research is aimed at tackling the following objectives: (1) capture shocks under compressible conditions, (2) track the phase interface using either a SIM or DIM method, (4) incorporate surface tension effects, (5) incorporate viscous effects, (6) simulate the interaction

of shocks with immiscible interfaces, (7) utilize adaptive mesh refinement (AMR) techniques to provide high resolution in regions of interest, and (8) simulate the primary atomization of a liquid jet in a supersonic crossflow.

The remainder of this dissertation is organized as follows: Chapter 2 is a paper published in the Journal of Computational Physics, which describes a novel sharp interface approach using an in-cell reconstruction method [157] and a wave propagation algorithm [92] to simulate immiscible interfaces in compressible flows. Chapter 3 describes an extension to the in-cell reconstruction method to allow for jumps in the equation of state, as required for gas/liquid flows. Chapter 4 is a paper to be submitted to the Journal of Computational Physics, which describes a diffuse interface method [28, 45] used in combination with a novel adaptive mesh refinement framework [7] to simulate compressible multiphase flows. Chapter 5 describes the application the primary atomization of a liquid in supersonic gas crossflow. Finally, Chapter 6 summarizes the research work performed and provides recommendations for future work.

Chapter 2

AN IN-CELL RECONSTRUCTION FINITE VOLUME METHOD FOR FLOWS OF COMPRESSIBLE IMMISCIBLE FLUIDS

Karthik Kannan[1], Dominic Kedelty[1], Marcus Herrmann[2]

A paper published in the Journal of Computational Physics [78]

2.1 Abstract

We present a hybrid capturing/tracking method for finite volume solvers of compressible flows involving two immiscible fluids that are described by equations of states for ideal gases. The approach is an extension of the two-dimensional level set based in-cell-reconstruction method originally proposed by Smiljanovski, Moser, and Klein [157] for deflagration waves to three-dimensional flows involving interfaces between two immiscible fluids with surface tension forces. The interface motion is captured by an extension to the conservative, un-split geometric volume-of-fluid technique of Owkes and Desjardins [119]. The resulting method is a sharp interface method that avoids time-step restrictions or merging/mixing rules due to cut-cells by using cell-face-aperture averaged waves to update the volume averaged states of

[1]Ph.D. student, Arizona State University

[2]Associate Professor, Arizona State University

cells containing the interface directly. The method furthermore avoids the use of any mixed states in discretization stencils, by reconstructing the pure fluid states in each mixed cell, explicitly enforcing the jump conditions across the interface. Simulations of compressible flows that involve the interaction of shocks with interfaces between immiscible fluids are performed to demonstrate the performance of the proposed method.

2.2 Introduction

Compressible flows involving interfaces between immiscible fluids are common in both nature and technical applications. They include Richtmyer-Meshkov type instabilities in confinement fusion and supernova explosions, cavitation processes in naval applications, and high speed fuel injection in modern diesel and gasoline engines. Of particular interest are cases where shocks interact with interfaces between immiscible fluids, since this can result in complex flow dynamics that can be harnessed to achieve a desired effect. The two fluids separated by the interface may be physically immiscible, e.g., two-phase or water/oil systems, or they may have to be modeled as being immiscible on the length and time scales that are of interest or resolvable in a simulation, e.g., in large scale gas/gas systems.

In the past two decades, a number of numerical methods have been developed to simulate interfaces between immiscible fluids in compressible flows, see for example Jemison, Sussman, and Arienti [72] for a recent overview. They may be classified

in a number of different ways, among them whether they account for a macroscopic boundary between the fluids with a single velocity field, or whether the fluids are thoroughly mixed and are allowed to possess different velocities [114]. The latter approach requires momentum exchange terms between the fluids that relax to the single velocity formulation in the limit of infinitely fast momentum relaxation [6, 137, 138, 113].

Methods able to treat the macroscopic fluid interface using a single velocity can further be classified into whether they maintain that fluid interface as a discontinuity, or whether the interface smears out over a number of computational cells either due to dissipative numerical errors, or by design, by imposing a smooth transition function between the fluids. In the latter case, "unphysical" mixed states between the pure fluid states develop that may result in unphysical pressure oscillations in the solution [1, 2]. To avoid these, the captured interface motion must be described by the solution of advection equations in non-conservative form for specific combinations of material properties of the fluids [1, 2, 133, 153, 151, 74, 108].

Methods able to maintain the fluid interface as a discontinuity are usually called sharp interface methods. They require some means of locating the fluid interface in the computational domain and then avoid dissipative errors across the interface by either solving the two-fluid Riemann problem at the fluid interface directly [25, 38] or by introducing so-called ghost states eliminating numerical stencils across the fluid interface discontinuity. The idea of ghost states first introduced by Glimm, Marchesin, and McBryan [48] and later extended by Fedkiw et al. [42] uses extrapolation from

fluid cells of the same type to replace stencil cells that would be filled by the fluid of the other type. The formulation by Fedkiw et al. [42] is based on an interface capturing level set finite difference approach, i.e., the computational algorithm solves for the fluid state at a discrete location, typically the cell center, instead of an average fluid state representative of the computational cell, as would be the case for a finite volume method. The front tracking method of Terashima and Tryggvason [161] is based on this finite difference approach [42] but solves an additional advection equation to extrapolate discontinuous variables into the ghost fluid regions. The formulation of Glimm, Marchesin, and McBryan [48] and Glimm et al. [51] on the other hand uses front tracking and maintains two states for cells containing the fluid interface, however, this may generate arbitrarily small sized cut cells that requires a merging algorithm of the cut cell with one of its neighbors.

A finite volume version of the ghost fluid method was introduced by Hu et al. [63] and Hu, Adams, and Iaccarino [62]. The method is in essence a hybrid of the method by Fedkiw et al. [42] and Glimm et al. [51] in that the interface motion is captured by a level set approach and two states are maintained for each cell containing the fluid interface using extrapolation to populate the required ghost states. As in [51], the method requires some merging/mixing rules to avoid small cut cells.

One challenge for ghost fluid methods is their application to shock impedance matching problems [169]. To address this challenge, Wang, Liu, and Khoo [169] proposed the level set based so-called real ghost fluid method. The method solves a Riemann problem at the fluid interface and uses its solution to populate ghost states

and modify interface adjacent real states. In two-dimensions, the method requires a dedicated algorithm to find an appropriate partner state for the Riemann problem solver and the solution of an additional extension PDE to populate ghost cells.

Bo and Grove [13] proposed a modification to the real ghost fluid method coupling the method to a volume of fluid approach to arrive at a conservative method. However, the employed split volume-of-fluid advection requires a cleanup procedure that may impact local conservation properties in addition to removing small scale interface features.

A method that combines a relaxation scheme for pressure equilibrium in interface cells with a volume-of-fluid method for describing the interface motion was introduced by Colella, Glaz, and Ferguson [27] in an unpublished manuscript. It represents the fluid interface as a sharp discontinuity, but resorts to a so-called Simple Line Interface Calculation (SLIC) subgrid interface reconstruction [116] that suffers from significant amount of flotsam, i.e., erroneously generated small scale interface structures. Improving on the SLIC reconstruction method, Miller and Colella [110] employed the Piecewise Linear Interface Calculation (PLIC) reconstruction of the fluid interface geometry.

The method proposed by Nourgaliev, Dinh, and Theofanous [117] uses exact two-fluid Riemann solvers at the fluid interface employing so-called characteristic based matching to couple solutions across the fluid interface. The method uses level sets to describe the motion of the fluid interface. Its drawback is that it deals with small cutcells in a non-conservative manner.

Finally, the method by Jemison, Sussman, and Arienti [72] extends projection methods traditionally used for incompressible flows to compressible flows using a moment of fluid method to describe the interface motion.

The purpose of the present contribution is to propose an alternative to the above mentioned approaches that is based entirely on a finite volume approach, not only for the compressible Navier-Stokes equations, but also the interface capturing approach. A finite volume framework naturally leads to a volume-of-fluid (VoF) method to capture the interface between immiscible fluids, since the volume-of-fluid scalar is the cell average of an indicator function marking the interface location. The proposed method is a sharp interface method, i.e., it maintains the interface between the fluids as a discontinuity without any numerical or prescribed smearing of the jump conditions across the interface. The method furthermore enforces local conservation and avoids the small cut-cell problem by updating directly volume averaged states in each computational cell.

The proposed method is an extension of the in-cell-reconstruction hybrid tracking/capturing method originally proposed by Smiljanovski, Moser, and Klein [157] and Smiljanovski [156] for deflagration waves, using a level set method to describe the location and motion of the deflagration wave interface in two dimensions. Extensions to three dimensions were later propose by Schmidt and Klein [143], using again a level set technique to capture the deflagration wave. An extension to interfaces between immiscible fluids in two-dimensional flows was used by Herrmann, Moin, and Abarzhi [60] to simulate Richtmyer-Meshkov instabilites, using a level set technique and

neglecting the effect of surface tension between the fluids. The drawback of applying these level set based hybrid tracking/capturing methods to immiscible fluids is the fact that the level set method used to capture the interface cannot be constructed to be discretely conservative in the limit of solenoidal velocity fields. In cases where there is mass transfer through the interface, as is the case for deflagration waves, such errors manifest themselves as errors in deflagration wave speed, i.e., errors in the burning rate, a quantity that is typically modeled and thus prone to inherent modeling errors anyways. However, for immiscible fluids without mass transfer, such numerical conservation errors can accumulate and become prohibitive, requiring needlessly fine meshes to control. Combining the hybrid tracking/capturing scheme with an inherently conservative scheme in the limit of solenoidal velocity fields, like an un-split geometric VoF method, is thus desirable and one of the goals of this contribution.

This paper is structured as follows: the governing equations and jump conditions describing the flow of two compressible immiscible fluids are presented in Section 2.3. The numerical methods employed to solve these equations are described in Section 2.4. The results of test cases used to verify various aspects of the proposed numerical approach are presented in Section 2.5. Finally, the current work is concluded and potential areas of future work are outlined in Section 2.6.

2.3 Governing Equations

2.3.1 Single Fluid Formulation

The equations describing the motion of compressible fluids are the continuity, Navier-Stokes, and energy equations,

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \boldsymbol{u}) = 0 \tag{2.1a}$$

$$\frac{\partial \rho \boldsymbol{u}}{\partial t} + \nabla \cdot (\rho \boldsymbol{u}\boldsymbol{u}) + \nabla p = \nabla \cdot \overline{\overline{\tau}} \tag{2.1b}$$

$$\frac{\partial \rho E}{\partial t} + \nabla \cdot ((\rho E + p)\, \boldsymbol{u}) = \nabla \cdot \left(\overline{\overline{\tau}} \boldsymbol{u} - \boldsymbol{q}\right) \,, \tag{2.1c}$$

with ρ the density, $\boldsymbol{u} = (u, v, w)^T$ the velocity vector, p the pressure, $\overline{\overline{\tau}}$ the viscous stress tensor,

$$\overline{\overline{\tau}} = \mu \left(\nabla \boldsymbol{u} + (\nabla \boldsymbol{u})^T - \frac{2}{3} (\nabla \cdot \boldsymbol{v}) \overline{\overline{I}} \right) \,, \tag{2.2}$$

with μ the dynamic viscosity and $\overline{\overline{I}}$ the identity matrix, E the specific total energy,

$$E = i + \frac{1}{2} \boldsymbol{u} \cdot \boldsymbol{u} \,, \tag{2.3}$$

with i is the specific internal energy, and $\boldsymbol{q}$ the heat flux vector,

$$\boldsymbol{q} = -k \nabla T \,, \tag{2.4}$$

with k the coefficient of thermal conductivity, and T the temperature.

The above system of equations is closed by a law for the dynamic viscosity, $\mu = \mu(T)$, and an equation of state, $p = p\,(\rho, i)$. Here, we use Sutherland's formula [160],

$$\frac{\mu}{\mu_0} = \left(\frac{T}{T_0}\right)^{3/2} \frac{T_0 + S}{T + S}\,, \tag{2.5}$$

where μ_0 denotes the dynamic viscosity at a reference temperature T_0, and S is the Sutherland constant, and the equation of state for an ideal gas,

$$p = (\gamma - 1)\ \rho i\,, \tag{2.6}$$

with γ the ratio of specific heats. This results in the speed of sound c as

$$c = \sqrt{\gamma\,\frac{p}{\rho}} \tag{2.7}$$

and the temperature T being

$$T = \frac{p}{(\gamma - 1)\ c_v\,\rho}\,, \tag{2.8}$$

with c_v the specific heat at constant volume.

2.3.2 Immiscible Two-Fluid Formulation

The compressible flow of two immiscible fluids 0 and 1 is described by Eqs. (2.1a) to (2.1c) within each fluid. At the interface Γ separating the two fluids, the states of the two fluids are coupled by jump conditions, denoted by $[\![\cdot]\!]_\Gamma$, that can be derived from mass, momentum, and energy conservation across the interface, see for example

[141],

$$[\![\mathbf{u}]\!]_\Gamma = 0 \tag{2.9a}$$

$$[\![p\overline{\overline{I}} - \overline{\overline{\tau}}]\!]_\Gamma \cdot \boldsymbol{n} = \sigma\kappa\boldsymbol{n} \tag{2.9b}$$

$$[\![T]\!]_\Gamma = 0\,, \tag{2.9c}$$

with σ the surface tension coefficient, and κ and $\boldsymbol{n}$ the curvature and normal vector of the interface Γ. Note that Eq. (2.9a) assumes a no-slip condition for velocity at the interface and hence a zero-jump of both normal and tangential velocity components, with the latter valid only for the viscous fluids studied here. Similarly, Eq. (2.9c) assumes continuity of temperature, valid only for non-zero thermal conductivity fluids. The location of Γ can be captured by an indicator function C,

$$C(\boldsymbol{x}, t) = \begin{cases} 1 & \text{if} \quad \boldsymbol{x} \in \Omega_1 \\ 0 & \text{if} \quad \boldsymbol{x} \in \Omega_0 \end{cases} \tag{2.10}$$

where Ω_1 and Ω_0 are the volumes occupied by fluid 1 and fluid 0, respectively.

2.4 Numerical Methods

In this section, we present the hybrid tracking/capturing in-cell-reconstruction algorithm for structured Cartesian meshes. Extensions to unstructured meshes are in principle possible, but beyond the scope of the current paper. First, the finite-volume method used to solve the single fluid equations are briefly reviewed. Next, the extension to two immiscible fluids is discussed and the in-cell-reconstruction scheme

for reconstructing pure fluid states in mixed cells is presented. Finally, extensions to the un-split geometric volume-of-fluid method of Owkes and Desjardins [119] are presented that can be used not only in the context of the here presented hybrid capturing/tracking method to transport the interface as a wave, but also enable geometric un-split transport of volume-of-fluid scalars an arbitrary unstructured meshes.

2.4.1 Single Fluid Finite Volume Method

The proposed solution method is based on a finite volume discretization of the governing equations, Eqs. (2.1a) to (2.1c). For each control volume Ω_{cv}, the volume average of the conservative variables $\boldsymbol{Q} = (\rho, \rho u, \rho v, \rho w, \rho E)^T$ is stored and solved for,

$$\boldsymbol{Q}_{cv} = \frac{1}{V_{cv}} \int_{\Omega_{cv}} \boldsymbol{Q}(\boldsymbol{x})\, d\boldsymbol{x} \tag{2.11}$$

where, V_{cv} is the volume of Ω_{cv}. The left-hand-side of Eqs. (2.1a) to (2.1c), i.e., the compressible Euler equations, are solved using the second-order in space finite-volume wave propagation algorithm of LeVeque [90], briefly summarized in the following. For a comprehensive review of the single fluid method used here, the interested reader is referred to the textbook of LeVeque [92] and the references cited therein.

Using a Godunov approach, local cell face Riemann problems can be defined using as left and right states, the control volume averages of $\boldsymbol{Q}$ in the two cells sharing the face, i.e., $\boldsymbol{Q}_{cv}^L$ and $\boldsymbol{Q}_{cv}^R$. The cell face Riemann problems are linearized using Roe

averages [136, 47] such that the jump in $\boldsymbol{Q}$ across the cell face is decomposed into

$$\boldsymbol{Q}_{cv}^{L} - \boldsymbol{Q}_{cv}^{R} = \sum_{p=1}^{m} \alpha_p \boldsymbol{r}_p = \sum_{p=1}^{m} \mathcal{W}^p \,, \tag{2.12}$$

where $\boldsymbol{r}_p$ is the p-th eigenvector of the Roe matrix [136], α_p is the wave strength of the p-th wave, and $\mathcal{W}^p = \alpha_p \boldsymbol{r}_p$ is the p-th wave [92]. The left (-) and right (+) going fluctuations $\mathcal{A}^{\pm}\Delta\boldsymbol{Q}$ can then be determined from

$$\mathcal{A}^{\pm}\Delta\boldsymbol{Q} = \sum_{p=1}^{m} (\lambda^p)^{\pm} \mathcal{W}^p \,, \tag{2.13}$$

where λ^p is the eigenvalue of the p-th eigenvector $\boldsymbol{r}_p$, and $(\lambda^p)^+ = \max(0, \lambda^p)$ and $(\lambda^p)^- = \min(0, \lambda^p)$, subjected to the entropy fix by Harten and Hyman [58], see LeVeque [92] for implementation details.

The update for the control volume conservative variables $\boldsymbol{Q}_{cv}$ due to the convective terms using the wave scheme in, for example 1D, then is (see Eq. (15.62) on page 329 and its derivation in LeVeque [92]).

$$\mathbf{Q}_{cv}^{n+1} = \mathbf{Q}_{cv}^{n} - \frac{\Delta t}{\Delta x}\left(\mathcal{A}^{-}\Delta\boldsymbol{Q}_{i+1/2} + \mathcal{A}^{+}\Delta\boldsymbol{Q}_{i-1/2}\right) - \frac{\Delta t}{\Delta x}\left(\tilde{\mathbf{F}}_{i+1/2} - \tilde{\mathbf{F}}_{i-1/2}\right) \tag{2.14}$$

where subindex $i \pm 1/2$ indicates the left, respective right cell face of the control volume, and $\tilde{\boldsymbol{F}}$ is a correction flux to achieve second-order accuracy in space (see Eq. (15.63) on page 330 and its derivation in LeVeque [92]),

$$\tilde{\mathbf{F}} = \frac{1}{2}\sum_{p=1}^{m} |\lambda^p| \left(1 - \frac{\Delta t}{\Delta x}|\lambda^p|\right) \tilde{\mathcal{W}}^p \tag{2.15}$$

with $\tilde{\mathcal{W}}^p$ being a limited wave using the monotonized central-difference limiter (MC limiter) of Leer [89].

In two and three dimensions, a dimensionally split version of the wave distribution algorithm using first-order in time Godunov splitting is employed [92].

The viscous terms on the right-hand-side of Eqs. (2.1b) to (2.1c) are incorporated using a simple first order in time splitting approach. The viscous stress tensor $\overline{\overline{\tau}}$ and heat flux vector $\boldsymbol{q}$ are computed at cell faces to calculate the viscous and diffusive cell face fluxes,

$$\Delta(\rho \boldsymbol{u})_{cv}^{f} = \overline{\overline{\tau}}_{f} \tag{2.16a}$$

$$\Delta(\rho E)_{cv}^{f} = \left(\overline{\overline{\tau}}\boldsymbol{u} - \boldsymbol{q}\right)_{f} \tag{2.16b}$$

For example, for face-normal components, such as τ_{xx} and q_x at $x-$normal control volume faces, the required values are calculated directly from the face adjacent control volume values using central differences. However, for tangential derivatives such as those arising in $\overline{\overline{\tau}}$, the values are calculated first at the face-adjacent control volumes using central differences of the tangential control volume values and then averaged to the cell faces.

As an example, Fig. 2 shows the stencil required to calculate the y-direction derivative for a two-dimensional grid. First, the values at $(i, j+1)$ and $(i, j-1)$ are used to calculate the derivative at (i, j), and similarly values at $(i+1, j+1)$, $(i+1, j-1)$ are used to calculate the derivative at $(i+1, j)$. Then the required value at the face location $(i+1/2, j)$, is an arithmetic average of the values at (i, j) and $(i+1, j)$.

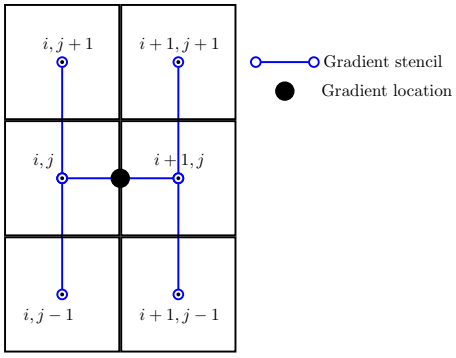


Figure 2: Stencil requirements to calculate $y-$direction derivative at the $x-$normal face

2.4.1.1 Single Fluid Time Step Restriction

The solution is advanced in time using an explicit first-order forward Euler method, subject to the following time step stability constraint

$$\Delta t = \text{CFL} \times \min\left(\Delta t_{conv}, \Delta t_{visc}\right) \tag{2.17}$$

with

$$\Delta t_{conv} = \min_{cv}\left(\frac{\min(\Delta x, \Delta y, \Delta z)}{(\max(|u|, |v|, |w|) + c)}\right) \tag{2.18}$$

and

$$\Delta t_{visc} = \min_{cv} \frac{\min(\Delta x, \Delta y, \Delta z)^2}{\mu} \tag{2.19}$$

where all quantities are specific to each control volume.

2.4.2 Immiscible Two-Fluid Finite Volume Method: Hybrid Tracking/Capturing

Applying the definition of the control volume average of finite volume methods to the indicator function C, Eq. (2.10), results in the definition of the fluid 1 volume fraction α,

$$\alpha = \frac{1}{V_{cv}} \int_{\Omega_{cv}} C(\boldsymbol{x})\, d\boldsymbol{x}\,. \tag{2.20}$$

Since the interface is a material surface, the advection equation for α is

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot (\boldsymbol{u}\alpha) = \alpha \nabla \cdot \boldsymbol{u}\,. \tag{2.21}$$

The two fluid algorithm thus requires solution not only of Eqs. (2.1a) to (2.1c), but also Eq. (2.21). Unfortunately, the single fluid algorithm described in Section 2.4.1 to solve Eqs. (2.1a) to (2.1c) is applicable only if neither control volume adjacent to the cell face of interest contains the interface, i.e., both the left and right states of the Riemann problem $\boldsymbol{Q}_{cv}^{L}$ and $\boldsymbol{Q}_{cv}^{R}$ are the respective pure fluid 0 or fluid 1 states. However, should either of the cells contain the interface, the single fluid cell face Riemann problem approach would involve averaged states of both fluids, resulting in a non-sharp interface treatment. To maintain a sharp interface description, we first reconstruct the pure fluid states from the cell averages of the cells containing the interface, see Section 2.4.2.1, use these reconstructed states to construct cell face Riemann problems within each fluid separately, and then use these single fluid cell face waves to determine cell face average wave contributions to the mixed cell

updates, see Section 2.4.2.2. In a wave based algorithm, the interface itself constitutes a wave. The conservation property of the overall scheme in interface cells thus depends directly on the accuracy of solving Eq. (2.21). To avoid any artificial smearing of the interface wave, we solve Eq. (2.21) geometrically, using an unsplit VoF scheme described in Section 2.4.3. The interface curvature is calculated using the mesh-decoupled height function method of Owkes and Desjardins [120]. The approach to determine the interface wave contribution to the control volume update is summarized in Section 2.4.2.3.

2.4.2.1 In-Cell Reconstruction

The goal of in-cell-reconstruction is to avoid mixed-fluid Riemann problems and set up Riemann problems that are purely within a single fluid [157]. Let $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_0$ be the respective pure fluid 1 and fluid 0 states in the control volume Ω_{cv} containing part of the interface, then following Eq. (2.11),

$$\boldsymbol{Q}_{cv} = \frac{1}{V_{cv}} \int_{\Omega_{cv}} \mathbf{Q}(\mathbf{x})\, d\mathbf{x} = \frac{1}{V_{cv}} \left(\int_{\Omega_1} \mathbf{Q}_1(\mathbf{x})\, d\mathbf{x} + \int_{\Omega_0} \mathbf{Q}_0(\mathbf{x})\, d\mathbf{x} \right) \tag{2.22}$$

where $\Omega_{cv} = \Omega_1 + \Omega_0$ with Ω_0 the cell volume occupied by fluid 0 and Ω_1 the cell volume occupied by fluid 1. Denoting by $\boldsymbol{Q}_{cv,1}$ and $\boldsymbol{Q}_{cv,0}$ the averages of $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_0$ over Ω_1 respective Ω_0, and using Eq. (2.20), results in

$$\boldsymbol{Q}_{cv} = \alpha\, \boldsymbol{Q}_{cv,1} + (1 - \alpha)\, \boldsymbol{Q}_{cv,0} \tag{2.23}$$

As shown in Fig. 3, if one were able to reconstruct $\boldsymbol{Q}_{cv,1}$ and $\boldsymbol{Q}_{cv,0}$ from the known cell average $\boldsymbol{Q}_{cv}$, one could solve cell face Riemann problems in either fluid separately, avoiding Riemann problems involving the mixed state $\boldsymbol{Q}_{cv}$.

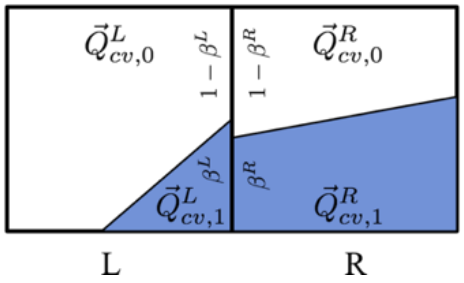


Figure 3: Cell face Riemann problems using reconstructed states

To derive the equations for reconstructing the primitive variables in both fluids from the cell averaged state $\boldsymbol{Q}_{cv} = (\overline{\rho}, \overline{\rho u}, \overline{\rho v}, \overline{\rho w}, \overline{\rho E})^T$, we combine Eq. (2.23) with the jump conditions across the interface, Eqs. (2.9a) to (2.9c), resulting in

$$
\begin{aligned}
u_0 &= u_1 = \frac{\overline{\rho u}}{\overline{\rho}}, \quad v_0 = v_1 = \frac{\overline{\rho v}}{\overline{\rho}}, \quad w_0 = w_1 = \frac{\overline{\rho w}}{\overline{\rho}} \\
p_0 &= \frac{(\gamma_0 - 1)(\gamma_1 - 1)\left(\overline{\rho E} - \frac{1}{2}\rho\,||\boldsymbol{u}||^2\right)}{\alpha(\gamma_0 - 1) + (1-\alpha)(\gamma_1 - 1)} - \frac{\alpha(\gamma_0 - 1)\left(\sigma\,\kappa + [\![\boldsymbol{n}^T \cdot \overline{\overline{\tau}} \cdot \boldsymbol{n}]\!]_\Gamma\right)}{\alpha(\gamma_0 - 1) + (1-\alpha)(\gamma_1 - 1)} \\
p_1 &= p_0 + \sigma\,\kappa + [\![\boldsymbol{n}^T \cdot \overline{\overline{\tau}} \cdot \boldsymbol{n}]\!]_\Gamma \\
\rho_0 &= \frac{\overline{\rho}\,p_0\,(\gamma_1 - 1)\,c_{v,1}}{\alpha\,p_1\,(\gamma_0 - 1)\,c_{v,0} + (1-\alpha)\;p_0\,(\gamma_1 - 1)\,c_{v,1}} \\
\rho_1 &= \frac{\overline{\rho}\,p_1\,(\gamma_0 - 1)\,c_{v,0}}{\alpha\,p_1\,(\gamma_0 - 1)\,c_{v,0} + (1-\alpha)\;p_0\,(\gamma_1 - 1)\,c_{v,1}}
\end{aligned}
\tag{2.24}
$$

with $||\boldsymbol{u}||^2 = u_0^2 + v_0^2 + w_0^2 = u_1^2 + v_1^2 + w_1^2$. These equations require the viscosity in each fluid to calculate the $[\![\boldsymbol{n}^T \cdot \overline{\overline{\tau}} \cdot \boldsymbol{n}]\!]_\Gamma$ term. For non-temperature dependent viscosities, i.e., $\mu_0 = \text{const}$ and $\mu_1 = \text{const}$, this is straight-forward. However, if the viscosity is temperature dependent, see Eq. (2.5), the fluid temperatures $T_0 = T_1 = T$, see Eq. (2.9c), are required, resulting in the following non-linear equation for T,

$$\begin{aligned} A_0\, T + A_1\, \frac{\mu_{\text{ref},1}\,(T_{\text{ref},1} + S_1)}{T_{\text{ref},1}^{3/2}}\, \frac{T^{3/2}}{T + S_1}\, \boldsymbol{n}^T \cdot \left(\nabla \boldsymbol{u}_1 + (\nabla \boldsymbol{u}_1)^T - \frac{2}{3}\,(\nabla \boldsymbol{u}_1)\,\overline{\overline{I}}\right) \cdot \boldsymbol{n} \\ - A_1\, \frac{\mu_{\text{ref},0}\,(T_{\text{ref},0} + S_0)}{T_{\text{ref},0}^{3/2}}\, \frac{T^{3/2}}{T + S_0}\, \boldsymbol{n}^T \cdot \left(\nabla \boldsymbol{u}_0 + (\nabla \boldsymbol{u}_0)^T - \frac{2}{3}\,(\nabla \boldsymbol{u}_0)\,\overline{\overline{I}}\right) \cdot \boldsymbol{n} \end{aligned} \tag{2.25}$$

$$- A_2 - A_3 = 0$$

where S_1, S_0 are the Sutherland constants for the individual fluids and A_0 through A_3 are given by

$$\begin{aligned} A_0 &= \bar{\rho}\, c_{v,0}\, c_{v,1}\, (\gamma_0 - 1)\, (\gamma_1 - 1) \\ A_1 &= \frac{\alpha\,(1-\alpha)\,(\gamma_0 - 1)\,(\gamma_1 - 1)\,(c_{v,1} - c_{v,0})}{\alpha\,(\gamma_0 - 1) + (1-\alpha)\,(\gamma_1 - 1)} \\ A_2 &= \alpha\,(\gamma_0 - 1)\, c_{v,0}\, \sigma\kappa \\ A_3 &= \frac{(\gamma_0 - 1)\,(\gamma_1 - 1)\left(\overline{\rho E} - \frac{1}{2}\rho\,||\boldsymbol{u}||^2\right) - \alpha\,(\gamma_0 - 1)\,\sigma\kappa}{\alpha\,(\gamma_0 - 1) + (1-\alpha)\,(\gamma_1 - 1)} \\ &\qquad \left(\alpha\,(\gamma_0 - 1)\, c_{v,0} + (1-\alpha)\,(\gamma_1 - 1)\, c_{v,1}\right) \end{aligned} \tag{2.26}$$

Equation (2.25) is solved using standard root finding techniques such as Newton's method or zeroin. Using the reconstructed fluid 1 and fluid 0 primitive variables obtainable from Eq. (2.24), the reconstructed conservative states in each fluid, $\boldsymbol{Q}_{cv,1}$ and $\boldsymbol{Q}_{cv,0}$, can then be calculated.

2.4.2.2 Mixed Cell Updates

Using the reconstructed individual fluid states in mixed cells, cell face Riemann problems within individual pure fluids can be constructed and solved. However, using the wave fluctuations from these Riemann problems to update the reconstructed states $\boldsymbol{Q}_{cv,1}$ associated with the fluid 1 cell volume Ω_1 and $\boldsymbol{Q}_{cv,0}$ associated with the fluid 0 cell volume Ω_0 directly would result in a small cut-cell time step limitation that is prohibitive, since α can be arbitrarily close to 0 or 1, and hence the associated volumes Ω_0 and Ω_1 can be arbitrarily small. Instead, only the cell-averaged mixed state $\boldsymbol{Q}_{cv}$ is updated [157].

To calculate the required average wave fluctuations $\mathcal{A}^{\pm}\Delta\boldsymbol{Q}_{cv}$ for each face intersected by the interface, the individual pure fluid 0 and fluid 1 wave fluctuations resulting from the pure fluid Riemann problems are averaged using the fluid 1 cell face area fraction β,

$$\mathcal{A}^{\pm}\Delta\boldsymbol{Q}_{cv} = \beta\mathcal{A}^{\pm}\Delta\boldsymbol{Q}_{cv,1} + (1-\beta)\,\mathcal{A}^{\pm}\Delta\boldsymbol{Q}_{cv,0}\,. \tag{2.27}$$

If the interface between cells is not continuous, as depicted in Fig. 3, in addition to the pure fluid Riemann problems, a mixed fluid Riemann problem arises. For well resolved interfaces, i.e., interfaces with local radii of curvature larger than the local mesh spacing, the cell face area fraction associated with the mixed Riemann problem is small, and we will approximate its contribution by an incomplete two fluid Riemann solver using only the contact wave, with the left and right states approximated using the wave-fluctuations of the single fluid Riemann problems. To achieve this, the fluid 1 face area fraction β used in Eq. (2.27) is calculated as the average of the left and right cells' values, $\beta = \left(\beta^L + \beta^R\right)/2$, and the interface segment coinciding with the cell face is transported geometrically using the VoF method described in Section 2.4.3. Approaches to calculate β^L and β^R from the interface geometry in the left and right cells are described in Section 2.4.3.

An important consequence of using average wave fluctuations calculated from single fluid Riemann problems, Eq. (2.27), to directly update the cell averages of mixed cells is the inherent assumption that the single fluid waves originating from cell faces do not further interact with the interface itself within a single time step. This is consistent with the linearization proposed by LeVeque and Shyue [94] for wave interactions and is in essence the large time step generalization of LeVeque [91] applied to wave/interface interactions only. Numerical tests, however, have shown that this linearization is viable only if the two fluids on either side of the interface are of comparable impedance, such as gas/gas systems. For liquid/gas systems,

the interaction of waves originating from the cell face Riemann problems with the interface has to be taken into account, but such an approach is beyond the scope of the current paper.

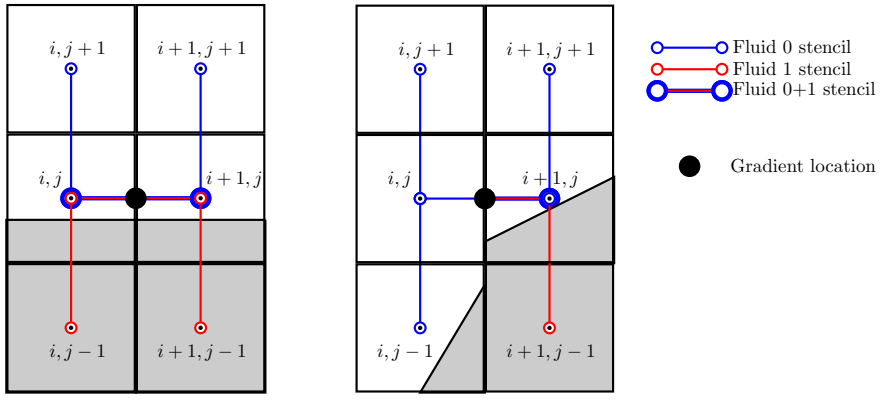
The above procedure for updating the mixed cells applies to the inviscid parts of Eqs. (2.1a) to (2.1c). For the viscous and diffusive terms a similar approach is taken. Instead of updating the pure fluid volume portions of the cell, a cell face average of the viscous and diffusive fluxes is calculated to update the mixed state of the cell directly,

$$\Delta(\rho \boldsymbol{u})^f_{cv} = \beta \, \overline{\overline{\tau}}_{1,f} + (1-\beta) \, \overline{\overline{\tau}}_{0,f} \tag{2.28}$$

$$\Delta(\rho E)^f_{cv} = \beta \, \left(\overline{\overline{\tau}}_1 \boldsymbol{u}_1 - q_1\right)_f + (1-\beta) \left(\overline{\overline{\tau}}_0 \boldsymbol{u}_0 - q_0\right)_f \, . \tag{2.29}$$

This necessitates that velocity and temperature gradients for both fluids are calculated at each cell face, although the value of fluid 0 gradients is irrelevant for faces with $\beta = 1$, i.e., pure fluid 1 faces, and the value of fluid 1 gradient is irrelevant for faces with $\beta = 0$, i.e., pure fluid 0 faces.

To ensure that gradients are calculated using only same fluid stencil points, the single fluid approach described in Section 2.4.1 is modified to revert to one-sided first order derivatives instead of second-order central derivatives if a stencil cell is filled entirely with the opposing fluid. Similarly, the averaging of cell center tangential gradients to cell faces is replaced by one-sided constant extrapolation from the cell center to the face, if a valid same fluid derivative exists only on one side of the cell face.



(a) Simple case **(b)** Extreme case

Figure 4: Stencil requirements to calculate $y-$direction derivatives near mixed cells

As an example, consider the 2-D interface geometries depicted in Fig. 4 for calculating values at the x-normal face between cells (i, j) and $(i+1, j)$. For a simple case of a horizontal interface as shown in Fig. 4a, the $y-$direction derivative of a fluid 0 quantity is calculated at (i, j) using one-sided finite-differences of fluid 0 values at (i, j) and $(i, j+1)$. Similarly, the derivative at $(i+1, j)$ is calculated using fluid 0 values at $(i+1, j)$ and $(i+1, j+1)$. These values are then averaged to obtain the derivative at the cell face location. The above stencil is mirrored about (i, j) to calculate the respective derivatives for fluid 1.

For a more extreme case as shown in Fig. 4b, the stencil is chosen as follows. For fluid 0, the derivative at (i, j) can be calculated using a central stencil with $(i, j+1)$ and $(i, j-1)$. The derivative at $(i+1, j)$ however, is calculated using one-sided finite-differences of values at $(i+1, j+1)$ and $(i+1, j)$, since the cell at $(i+1, j-1)$ is entirely filled with fluid 1 and therefore no valid fluid 0 state exists in that cell. The

arithmetic average of the fluid 0 derivatives at (i, j) and $(i+1, j)$ then gives the fluid 0 derivative at the cell face location. However, no valid stencil can be constructed for the fluid 1 derivative at (i, j). Therefore, the face value of the fluid 1 derivative is set equal to the fluid 1 derivative at $(i+1, j)$ which is calculated using fluid 1 values at $(i+1, j)$ and $(i+1, j-1)$.

The consequence of restricting to one-sided differences and one-sided extrapolation from cell centers to faces in the vicinity of the interface for certain interface geometries is that the formal order of accuracy of the viscous and diffusive terms is first-order near the interface, while remaining second-order elsewhere. The benefit of this approach is that no derivatives are calculated using either mixed cell average values or values of the opposite fluid. While neither velocity nor temperature actually exhibit a jump across the interface, their gradients do, provided there's a non-zero jump in μ and/or k at the interface.

2.4.2.3 Interface Wave

In the wave propagation scheme, the fluid interface itself is a wave and must be taken into account. Since the interface is a contact discontinuity, the wave fluctuation of the interface wave is simply

$$\Delta \boldsymbol{Q} = \boldsymbol{Q}_1 - \boldsymbol{Q}_{\boldsymbol{0}} \,, \tag{2.30}$$

with $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_0$ the reconstructed fluid 1 and fluid 0 conservative variables, see Section 2.4.2.1. The motion of the interface wave is given by the solution to Eq. (2.21). We solve it using an extension of the geometric framework proposed by Owkes and Desjardins [119] described in Section 2.4.3. While the original geometric transport of Owkes and Desjardins [119] is based on the construction of unsplit cell face flux volumes that identify fluid 1 volumes that flux through a given cell face within a time step, here we need to identify individual interface wave fluctuations that are within the cell face flux volumes. The resulting change in conservative variables due to transport through the cell face f is

$$\begin{aligned}\Delta\boldsymbol{Q}^f_{cv} = \sum_n \frac{\left(\mathbf{Q}^n_1 - \mathbf{Q}^n_0\right)}{V_{cv}} \left[\alpha^n_{0,0}\Omega^n_{0,0} - \left(1-\alpha^n_{1,1}\right)\Omega^n_{1,1} + \alpha^n_{0,1}\Omega^n_{0,1} - \left(1-\alpha^n_{1,0}\right)\Omega^n_{1,0}\right] \\ - \frac{\sum_n \Omega^n_{0,1}}{V_{cv}}\left(\mathbf{Q}^R_{cv,1} - \mathbf{Q}^L_{cv,0}\right) + \frac{\sum_n \Omega^n_{1,0}}{V_{cv}}\left(\mathbf{Q}^L_{cv,1} - \mathbf{Q}^R_{cv,0}\right) ,\end{aligned} \tag{2.31}$$

where $\Omega^n_{0,0}$, $\Omega^n_{1,1}$, $\Omega^n_{1,0}$ and $\Omega^n_{0,1}$ are portions of the flux volume originating from the cell face portion that on either side of the face is fluid 0/fluid 0, fluid 1/fluid 1, fluid 1/fluid 0, or fluid 0/fluid 1, and are within a neighbor cell n see Section 2.4.3, $\alpha^n_{0,0}$, $\alpha^n_{1,1}$, $\alpha^n_{1,0}$ and $\alpha^n_{0,1}$ are the fluid 1 volume fractions in the respective flux volume portions and neighbor cell, and $\boldsymbol{Q}^n_1$ and $\boldsymbol{Q}^n_0$ are the reconstructed fluid 1 and fluid 0 conserved variables in the neighbor cell n, see for example Fig. 5. Note that $\Omega^n_{1,0}$ and $\Omega^n_{0,1}$ exist due to the discontinuous PLIC reconstruction of the interface geometry in the geometric VoF transport algorithm.

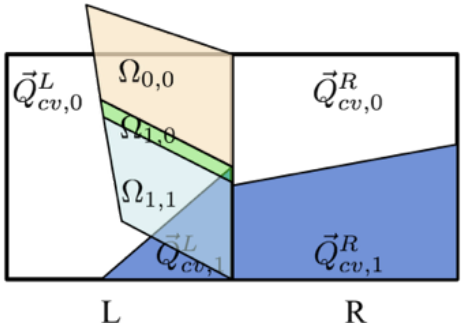


Figure 5: Flux volumes for interface wave construction

The sum in Eq. (2.31) has to extend over all possible neighbors n_f that could be intersected by the cell face f flux volumes, e.g., for the x-normal cell face shown in Fig. 5, this would be the $2 \times 3 \times 3$ cell stencil (in x, y, and z direction) centered on the cell face, provided the convective CFL number is no larger than unity.

Finally, the total change due to interface wave fluctuations for the control volume cv consists of the contributions from Eq. (2.31) summed over all faces f of the control volume, and the motion of the interface contained inside cv itself,

$$\Delta \boldsymbol{Q}_{cv} = (\boldsymbol{Q}_{cv,1} - \boldsymbol{Q}_{cv,0})\, \Delta\alpha^C_{cv} + \sum_{f_{cv}} \Delta \boldsymbol{Q}^f_{cv} \tag{2.32}$$

with

$$\Delta\alpha^C_{cv} = \Delta\alpha_{cv} - \frac{1}{V_{cv}} \sum_{f_{cv}} \sum_{n_f} \alpha^n_{0,0}\Omega^n_{0,0} + \alpha^n_{1,1}\Omega^n_{1,1} + \alpha^n_{1,0}\Omega^n_{1,0} + \alpha^n_{0,1}\Omega^n_{0,1} \tag{2.33}$$

and $\Delta\alpha_{cv}$ the change in α in control volume cv due to the geometric transport of α, see Section 2.4.3.

2.4.2.4 Immiscible Two-Fluid Time Step Restriction

The addition of surface tension forces in the jump conditions of pressure across the interface, see Eq. (2.9b), adds an additional stability time step constraint,

$$\Delta t_{cap} = \sqrt{\frac{(\min(\rho_1) + \min(\rho_0))\ \min(\Delta x, \Delta y, \Delta z)^3}{4\pi\sigma}}\,, \tag{2.34}$$

using the minima of the reconstructed densities in fluid 1, respective fluid 0. The overall time step is thus

$$\Delta t = \text{CFL} \times \min\left(\Delta t_{conv}, \Delta t_{visc}, \Delta t_{cap}\right) \tag{2.35}$$

with Δt_{conv} from Eq. (2.18) and

$$\Delta t_{visc} = \min_{cv} \frac{\min(\Delta x, \Delta y, \Delta z)^2}{\max(\mu_0, \mu_1)}\,. \tag{2.36}$$

In all cases presented in this paper, a CFL number of 0.5 is used.

2.4.3 Geometric Un-Split VoF Scheme for Interface Wave Transport

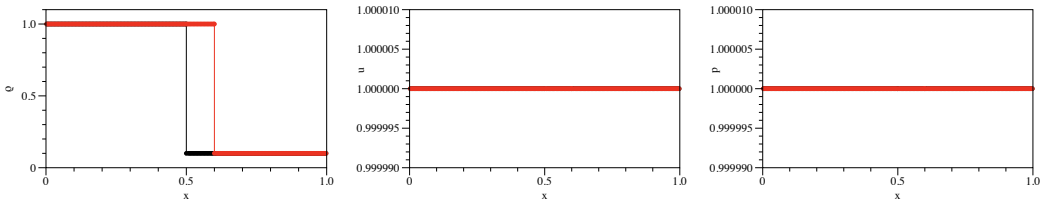
The interface tracking is done by an extension to the unsplit, conservative, and bounded method proposed by Owkes and Desjardins [119]. Since this section of the paper was contributed by the author Dominic Kedelty, who is a Ph.D. student at Arizona State University and is not a direct contribution of the presenting author, it is removed from this chapter altogether. However, for the sake of completeness, all details are provided in B.

2.5 Results

In this section test cases are presented to demonstrate the performance of the in-cell reconstruction method, ranging from three simple one-dimensional cases of inviscid advection, two fluid Riemann problem, and two fluid Couette flow, to two- and three-dimensional shock bubble interactions with and without surface tension.

2.5.1 One-Dimensional Advection of an Isolated Material Front

The first one-dimensional test case introduced by Abgrall and Karni [2] consists of a simple material front separating inviscid fluids with different densities and γ advected with a constant velocity. The front is initially placed in the center of a unit sized domain resolved by 200 equidistant cells, with values to the left of the material front of $(\rho, u, p, \mathrm{T}, \gamma, \mathrm{k})_L = (1, 1, 1, 1, 1.6, 0)$ and values to the right of $(\rho, u, p, \mathrm{T}, \gamma, \mathrm{k})_R = (0.1, 1, 1, 1, 1.4, 0)$.



(a) Density **(b)** Velocity **(c)** Pressure

Figure 6: Isolated material front advection at $t = 0$ (black) and 0.1(red) time units

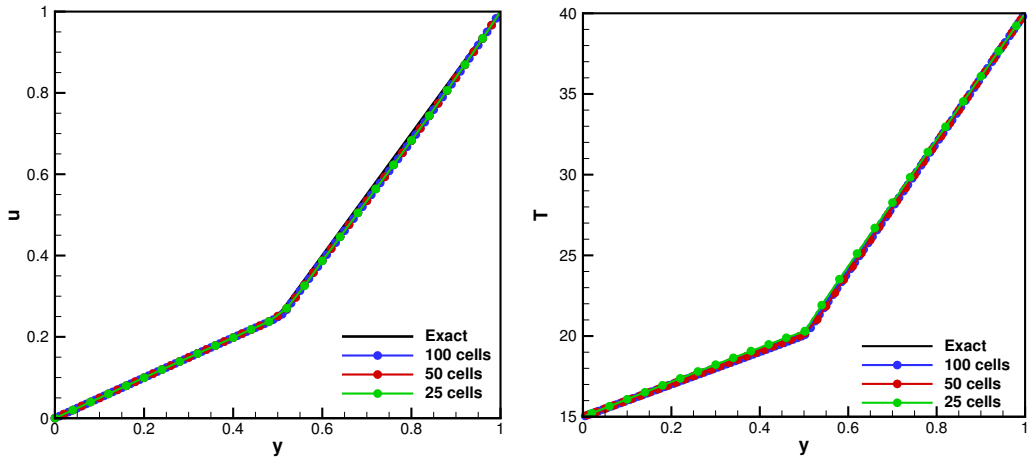
Figure 6 shows the density, velocity, and pressure at $t = 0$ and 0.1 time units. In cells containing the interface, both the reconstructed fluid states are shown at the location of the interface instead of the cell average value. As can be seen, the front is at the expected location, the front remains a discontinuity and no pressure or velocity oscillations present in standard conservative formulations [2] are observed.

2.5.2 One-Dimensional Two-Fluid Couette Flow

2.5.2.1 Velocity Test

The second test case is a two-fluid Couette flow between a stationary bottom wall and a unit velocity top wall with an interface placed a distance of $y = 0.502$ away from the bottom wall separating a bottom fluid with viscosity $\mu_0 = 0.1$ from a top fluid with viscosity $\mu_1 = 0.3$, assuming constant viscosity for each fluid. The distance between the walls is unity. Simulations are performed using 5 cells in the tangential wall direction with periodic boundary conditions and 25, 50, and 100 equidistant cells in the wall normal direction with no-slip conditions at the adiabatic walls. Both fluids are initially at rest and have unit density, temperature and pressure, constant Prandtl number of $\mathrm{Pr} = c_v \gamma \mu / k = 0.4$ and ratio of specific heats of 1.4.

Figure 7a shows the numerical solution of the tangential velocity as a function of the wall normal coordinate for different mesh resolutions compared to the exact solution at steady state. At the interface location, the change in velocity gradient due



(a) Tangential velocity profile **(b)** Temperature profile

Figure 7: Numerical solution to two-phase Couette flow compared to exact solution

to the jump in viscosity between the two fluids is clearly visible. Table 1 shows the infinity norm of the velocity error for the different mesh resolutions and the associated observed order of convergence. As expected, the observed order is one, due to the first-order treatment of the viscous terms at the interface.

2.5.2.2 Temperature Test

To verify that thermal effects are taken into account correctly, the test case is repeated with a stationary top wall at temperature $T_{w,t} = 40$, and a stationary bottom wall at $T_{w,b} = 15$. The constant viscosity of the top and bottom fluids are 0.2 and 0.8

respectively. Both fluids are initially at rest, have unit pressure and initialized with a uniform constant temperature of $T = 20$ and $\mathrm{Pr} = 10^{-4}$. All other fluid parameters are the same as in the velocity test in Section 2.5.2.1.

Table 1: Observed order of convergence for errors in velocity and temperature for the two-phase Couette flow test case

Cells in y	Velocity		Temperature	
	L_∞ norm	Order	L_∞ norm	Order
25	0.0472	-	0.3637	-
50	0.0325	1.0787	0.1647	1.1431
100	0.0251	1.0654	0.0607	1.4397

Figure 7b shows the numerical solution of the temperature as a function of the wall normal coordinate for different mesh resolutions compared to the exact solution. At the interface location, the change in temperature gradient due to the jump in thermal conductivity between the two fluids is clearly visible. Table 1 shows the infinity norm of the temperature error for different mesh resolutions and the associated observed order of convergence. As expected, the observed order is again one, due to the first-order treatment of the diffusive terms at the interface.

2.5.3 Verification of Surface-Tension Effects

2.5.3.1 Static Inviscid Drop with Prescribed Curvature

To verify the surface tension model, it is desirable to check if the surface tension forces discretely balance the pressure jump under static conditions. To this end the canonical test case of an inviscid column (in 2D) and sphere (in 3D) in equilibrium without gravity following the parameters suggested by Williams, Kothe, and Puckett [172] is preformed. A column (or sphere) of radius $R = 2$ is placed at the center of a 8^2 (or 8^3) domain with a constant grid spacing of $\Delta x = \Delta y = \Delta z = 0.2$. The exact curvature is prescribed. The surface tension coefficient σ is set to a value of 73. The pressure p_0 is set to unity and $p_1 = p_0 + \sigma\kappa$, following Eq. (2.9b). Both fluids are initially at rest with $T = 1$, $k = 0$, and $\gamma = 1.4$. The case is run for a single time step of size $\Delta t = 10^{-6}$ for varying density ratios (with unit ρ_0) and the errors in velocity and kinetic energy are computed.

As seen in Table 2, the errors in spurious currents are of the order of machine epsilon. Furthermore, the pressure jump across the interface is recovered exactly. Given the exact curvature, the algorithm achieves a discrete balance between surface tension and pressure jump across the interface even for high density ratios.

Table 2: Errors in velocity and kinetic energy after single time step for a static equilibrium drop using exact curvature

$\frac{\rho_1}{\rho_0}$	Column		Sphere	
	$L_\infty(\boldsymbol{u})$	E_{kin}	$L_\infty(\boldsymbol{u})$	E_{kin}
10^{-6}	2.01×10^{-20}	1.13×10^{-41}	4.02×10^{-20}	4.01×10^{-39}
10^{-3}	2.01×10^{-20}	1.17×10^{-41}	4.02×10^{-20}	4.06×10^{-39}
1	2.25×10^{-20}	5.48×10^{-41}	5.29×10^{-20}	1.15×10^{-38}
10^3	1.95×10^{-17}	1.40×10^{-38}	5.29×10^{-17}	2.68×10^{-36}
10^6	1.95×10^{-14}	1.35×10^{-35}	5.28×10^{-14}	2.55×10^{-33}

2.5.3.2 Long Time Evolution of Viscous Static Drop with Calculated Curvature

To further verify the modeling of surface tension in the present algorithm, simulations on the long time evolution of a viscous static drop at equilibrium are performed. In 2D, a column of diameter $D = 0.4$ is placed in the center of a unit sized box with uniform grid spacing of $h = 1/32$. The density ratio of the fluids is fixed as $\rho_1/\rho_0 = 1$, and by varying the density in both fluids the Laplace number, $\text{La} = 1/Oh^2 = \sigma\rho D/\mu^2$ is set. The Prandtl number is set at a constant $\text{Pr} = 0.7$ in both fluids. The capillary number, $Ca = |u_{\max}|\,\mu/\sigma$ at non-dimensional time $\tau = t\sigma/(D\mu) = 250$ is calculated for varying Laplace numbers. In all simulations, the viscosity in both fluids is $\mu = 0.1$, $\gamma = 1.4$, and $\sigma = 1$. The initial temperature in both fluids is $T = 1$ and the initial pressures are $p_0 = 1$ and $p_1 = p_0 + \sigma\kappa$, with κ set to the exact curvature of the column.

Table 3 shows the spurious current capillary number compared to Herrmann [61] using a level set method, Popinet and Zaleski [128] using a marker tracking method and Shin et al. [147] using a level contouring method, all of them being incompressible methods.

Table 3: Spurious current capillary number at $\tau = 250$ for varying Laplace number in the viscous equilibrium column test case

Dataset	La			
	1200	12,000	120,000	1,200,000
Present	1.81×10^{-6}	17.4×10^{-6}	19.36×10^{-6}	9.80×10^{-6}
[61]	0.12×10^{-6}	1.44×10^{-6}	3.09×10^{-6}	0.71×10^{-6}
[128]	5.99×10^{-6}	8.76×10^{-6}	-	-
[147]	2.18×10^{-6}	2.22×10^{-6}	-	-

The magnitude of the spurious currents in the present method is on the same order compared to other methods, albeit slightly higher. Table 4 shows a mesh refinement study of the spurious current capillary number for $\mathrm{La} = 12,000$ at non-dimensional time $\tau = 250$ compared to Herrmann [61] and Popinet and Zaleski [128]. The magnitude of the spurious currents converges under grid refinement with roughly second order, but again is slightly higher than in the two other incompressible methods.

Finally, Table 5 presents the L_∞ norm of the calculated curvature error at non-dimensional time $\tau = 250$ under grid refinement for the $\mathrm{La} = 12,000$ case. Note that $\tau = 250$ is more than twice the capillary time scale inherent in the problem, yet still

Table 4: Spurious current capillary number for La $= 12,000$ at $\tau = 250$ in the viscous equilibrium column test case under grid refinement

h	Present		[61]		[128]	
	Ca	Order	Ca	Order	Ca	Order
1/16	1.76×10^{-4}	-	4.92×10^{-6}	-	3.76×10^{-5}	-
1/32	1.74×10^{-5}	3.34	1.44×10^{-6}	1.77	6.68×10^{-6}	2.49
1/64	5.22×10^{-6}	1.74	0.34×10^{-6}	2.08	1.07×10^{-6}	2.64
1/128	1.50×10^{-6}	1.80	0.05×10^{-6}	2.77	0.12×10^{-6}	3.16

smaller than the viscous time scale [130, 131]. The observed order of convergence is two, consistent with the height function technique used to calculate curvature in the present method.

Table 5: L_∞ norm of the curvature error for La $= 12,000$ at $\tau = 250$ in the viscous equilibrium column test case under grid refinement

h	E_κ	Order
1/16	3.88×10^{-1}	-
1/32	9.48×10^{-2}	2.03
1/64	1.93×10^{-2}	2.30
1/128	5.17×10^{-3}	1.90

2.5.3.3 Zero Gravity Oscillation of a Column and Sphere

The final test case to verify the implementation of surface tension in the algorithm is the inviscid oscillation of a two-dimensional column and three-dimensional sphere. The column/sphere of radius $r = 2$ is placed in the center of a $[-10, 10]$ square/cube

box with slip boundary conditions on all sides and both fluids initially at rest. The column/sphere is initially perturbed by a mode $n = 2$ perturbation with an initial amplitude of $0.01r$. The theoretical oscillation period for two-dimensional columns in the linear regime is given by [85],

$$\omega^2 = \frac{n\left(n^2 - 1\right)\sigma}{\left(\rho_0 + \rho_1\right)\, r^3} \tag{2.37}$$

and for three-dimensional spheres,

$$\omega^2 = \frac{n\left(n^2 - 1\right)\left(n + 2\right)\sigma}{\left[\left(n + 1\right)\rho_1 + n\,\rho_0\right]\, r^3} \tag{2.38}$$

where the surface tension $\sigma = 1$, the density inside the drop $\rho_1 = 1$, and the density outside the drop $\rho_0 = 0.01$. In both fluids, $T = 1$, $\gamma = 1.4$, and $k = 0$. The initial pressures are $p_0 = 1$ and $p_1 = p_0 + \sigma\kappa$, with κ set to the calculated curvature of the closest interface cell. Table 6 shows the errors in oscillation period, $E_T = |T_{\text{calc}}\,\omega/2\pi - 1|$ for the oscillating column together with the results of Herrmann [61] and Torres and Brackbill [166]. The results obtained with the present method are comparable to those reported in the literature showing between first and second order convergence under mesh refinement.

Finally, Table 7 shows the errors in oscillation period for the three-dimensional sphere compared to the results reported in Herrmann [61]. The present method exhibits slightly larger error than the incompressible method, with observed orders of convergence between first and second order.

Table 6: Errors in oscillation period for two-dimensional column

h	E_T, column					
	Present	Order	[61]	Order	[166]	Order
20/64	6.02×10^{-2}	-	4.04×10^{-2}	-	13.2×10^{-2}	-
20/128	1.87×10^{-2}	1.69	1.05×10^{-2}	1.94	6.1×10^{-2}	1.11
20/256	0.61×10^{-2}	1.62	0.37×10^{-2}	1.50	1.5×10^{-2}	2.02

Table 7: Errors in oscillation period for three-dimensional sphere

h	E_T, sphere			
	Present	Order	[61]	Order
20/64	11.52×10^{-2}	-	8.50×10^{-2}	-
20/96	5.86×10^{-2}	1.67	3.85×10^{-2}	1.95
20/128	3.74×10^{-2}	1.56	2.08×10^{-2}	2.14
20/160	2.50×10^{-2}	1.81	1.19×10^{-2}	2.50

2.5.4 Two-Dimensional Shock-Bubble Interactions

This subsection presents results for the case of shocks in air interacting with circular bubbles of helium and R22 (Chlorodifluoromethane) initially at rest, following the experiments of Haas and Sturtevant [56]. Note that unlike in the experiments, the fluids here are treated as immiscible and no molecular mixing occurs across the interface. In all cases, a bubble of radius 0.025 is placed in the vertical center and 0.15 units from the left boundary of a rectangular two-dimensional domain of size 0.4×0.2. A planar shock of Mach number $\text{Ma} = 1.22$ is initialized 0.1 units from the

left boundary of the domain. Simple zero-order extrapolation non-reflective boundary conditions [92] are used on all boundaries. Unless otherwise stated, simulations are carried out using a mesh of 800×400 equidistant cells.

2.5.4.1 Shock in Air Impacting Helium Bubble

In the air shock/helium bubble case without surface tension, the pre-shock conditions in the air are set to $(\rho, u, p, \mathrm{T}, \gamma)_{air} = (1, 0, 1, 1, 1.4)$ and in the helium $(\rho, u, p, \mathrm{T}, \gamma)_{He} = (0.138, 0, 1, 1, 1.667)$. The viscosity and thermal conductivity of both the air and helium are calculated using Eq. (2.5) with $(T_0, \mu_0, S, \mathrm{Pr})_{air} = (275, 1.725 \times 10^{-5}, 110.9, 0.7)$ and $(T_0, \mu_0, S, \mathrm{Pr})_{He} = (273, 1.868 \times 10^{-5}, 79.9, 0.7)$.

Figure 8 shows numerical Schlieren images of the magnitude of the density gradient at various instances in time and Fig. 9 shows the temperature distribution at those times. It can be seen that the incoming shock is partially reflected and transmitted through the bubble, heating the helium bubble to a sightly higher temperature than the surrounding post-shock air due to the larger ratio of specific heats of helium as compared to air. At later stages, the bubble shows strong deformation with a central mushroom shaped stem of air penetrating the helium bubble and resulting in a thin region of helium at the downstream edge of the bubble.

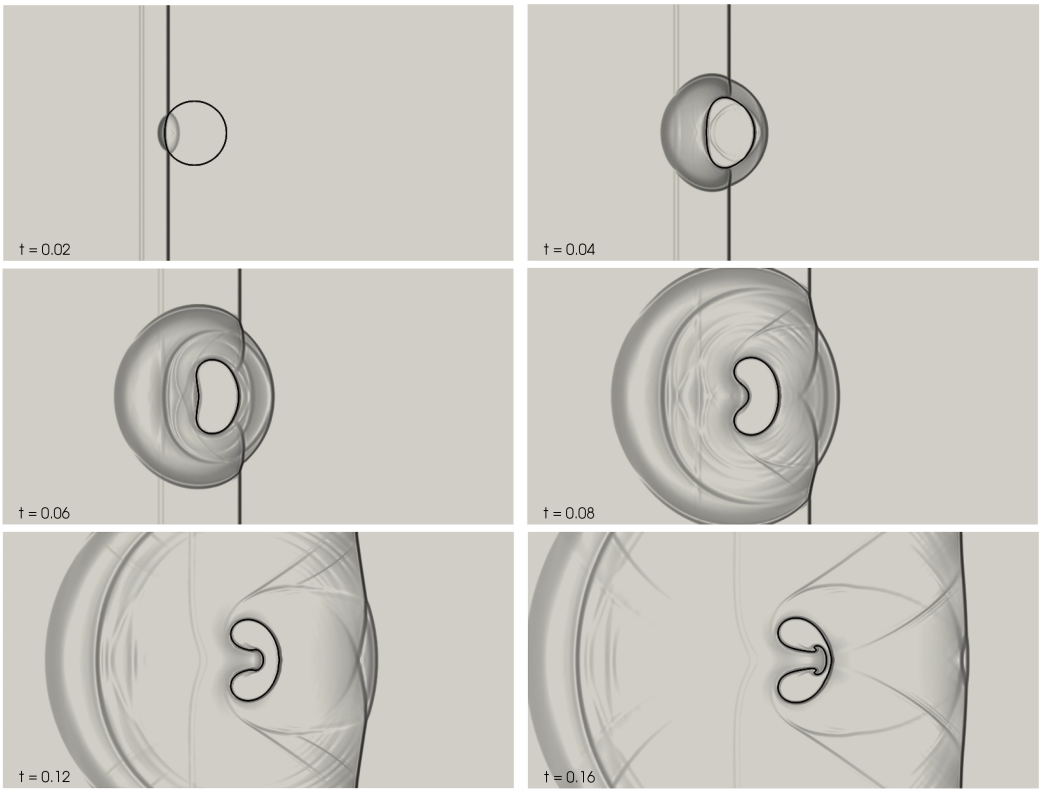


Figure 8: Numerical Schlieren images of Ma = 1.22 shock in air impacting a helium bubble (We = ∞) at $t = 0.02$, 0.04, 0.06, 0.08, 0.12, and 0.16 time units (from top left to bottom right). Log-scale for magnitude of density gradient ranging from 0.5 to 1000.

Figure 10 shows the PLIC reconstructed interface geometry at different times. At later times, the penetrating mushroom of air leads to an ever thinner sheet of helium that finally ruptures due to its thickness becoming smaller than the available local mesh resolution. Note that this topology change is purely numerical. Also, some secondary shear driven instabilities can be seen on the interface at the mushroom edges at later times. Note that these small scale dynamics are predicated on the two fluids being immiscible. In the experiments using miscible fluids, molecular mixing

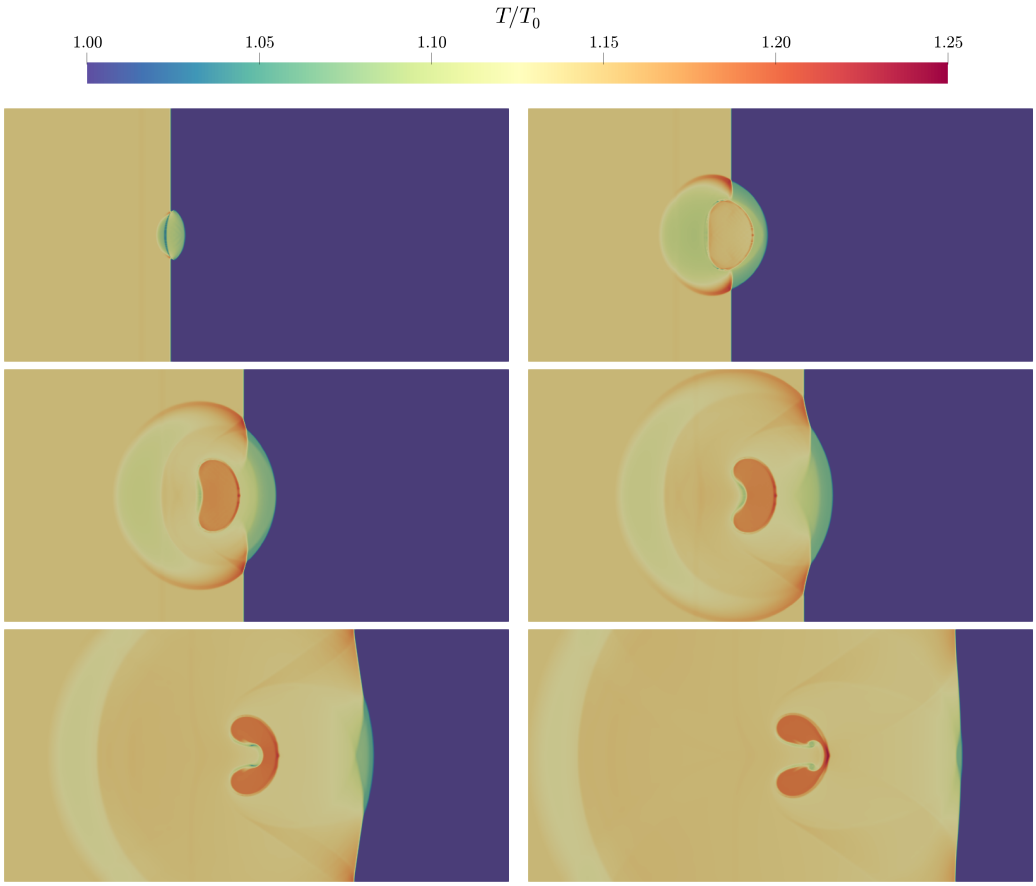


Figure 9: Temperature field normalized by pre-shock temperature of Ma = 1.22 shock in air impacting a helium bubble (We = ∞) at $t = 0.02$, 0.04, 0.06, 0.08, 0.12, and 0.16 time units (from top left to bottom right)

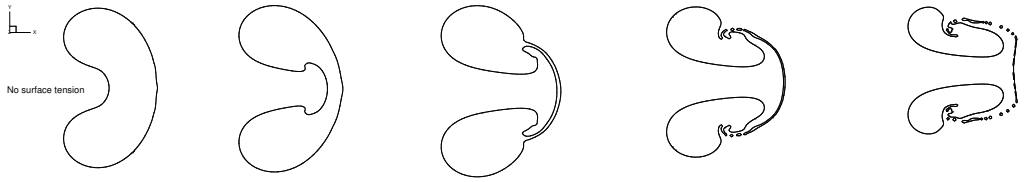
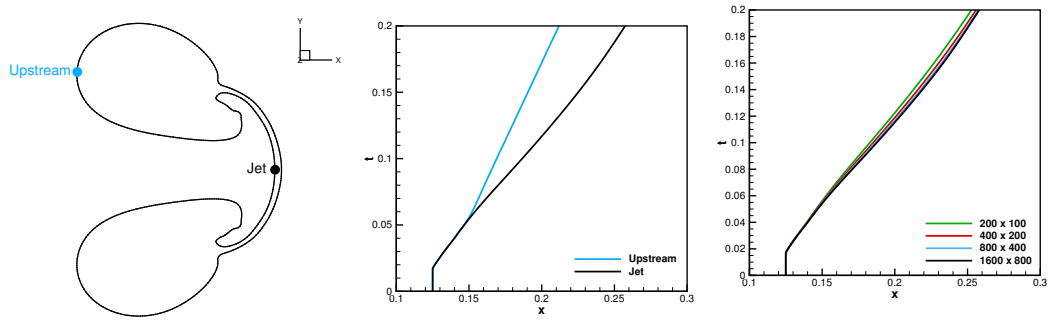


Figure 10: PLIC reconstructed interface geometries of the helium bubble at $t = 0.1,\ 0.15,\ 0.2,\ 0.25,\ 0.3$ (from left to right)

across the interface attenuates these small scale dynamics. Nonetheless, the interface shape is in qualitative good agreement to the experimentally observed interface shapes [56] for times when molecular mixing is not significant in the experiments.



(a) Interface position at $t = 0.2$

(b) Time history of interface positions

(c) Time history of jet interface position under mesh refinement

Figure 11: Mesh refinement study of air-helium interface position

Table 8: Grid Convergence Index (GCI) analysis of air-helium jet interface position at $t = 0.2$

Mesh	Interface position x	Observed order p	Richardson extrapolate x_0	GCI_{12} [%]	GCI_{23} [%]	Asymptotic range of convergence
200×100	0.2527	-	-	-	-	-
400×200	0.2559	-	-	-	-	-
800×400	0.2574	1.0426	0.2588	0.6993	1.4491	0.9941
1600×800	0.2580	1.2422	0.2585	0.2288	0.5428	0.9974

To verify that the solution converges under mesh refinement, both the position of the air/helium interface at the penetrating jet of air into the helium bubble and the most upstream position of the air/helium interface were recorded as a function of time, see Fig. 11. Table 8 shows the result of a Grid Convergence Index (GCI) analysis following the procedure of Roache [135] for the jet position x at $t = 0.2$. The Richardson extrapolated position of the jet using the finest mesh data of 1600x800 cells is $x = 0.2585 \pm 0.2288\%$ with an asymptotic range of convergence check value of 0.9974, very close to the theoretical value of 1, indicating the results are in the asymptotic range of grid convergence with an observed order of convergence slightly above one at 1.24.

2.5.4.2 Shock in Air Impacting Helium Bubble with Surface Tension

To demonstrate the impact of surface tension on the bubble dynamics, simulations were carried out for Weber numbers of $\mathrm{We} = \frac{\rho u^2 D}{\sigma} = 100$, 10, and 1, in addition to $\mathrm{We} = \infty$ of the previous section. In the definition of the Weber number, ρ and u are the post-shock density and velocity of air, and D is the diameter of the helium bubble. The initial and boundary conditions used are those of the prior section, however, the pre-shock pressure in the helium bubble is set to $p_{He} = 1 + \sigma\kappa$ to account for the presence of surface tension.

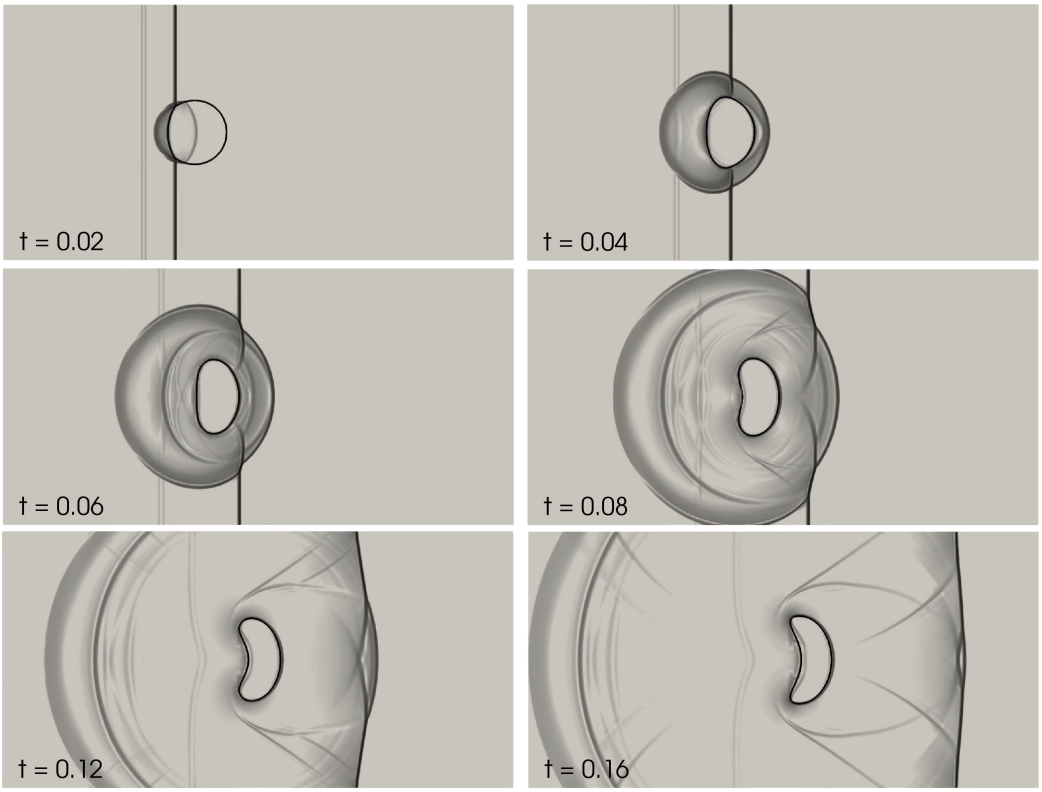


Figure 12: Numerical Schlieren images of Ma = 1.22 shock in air impacting a helium bubble (We = 10) at $t = 0.02$, 0.04, 0.06, 0.08, 0.12, and 0.16 time units (from top left to bottom right). Log-scale for magnitude of density gradient ranging from 0.5 to 1000.

Figure 12 shows numerical Schlieren images of the magnitude of the density gradient at various instances in time for the We = 10 case. While the dynamics of the refracted and reflected shock waves are similar to the We = ∞ case, see Fig. 8, the dynamics of the interface are markedly different. The presence of surface tension suppresses the formation of the jet of air penetrating the helium bubble, resulting in a crescent shape of the bubble at later times.

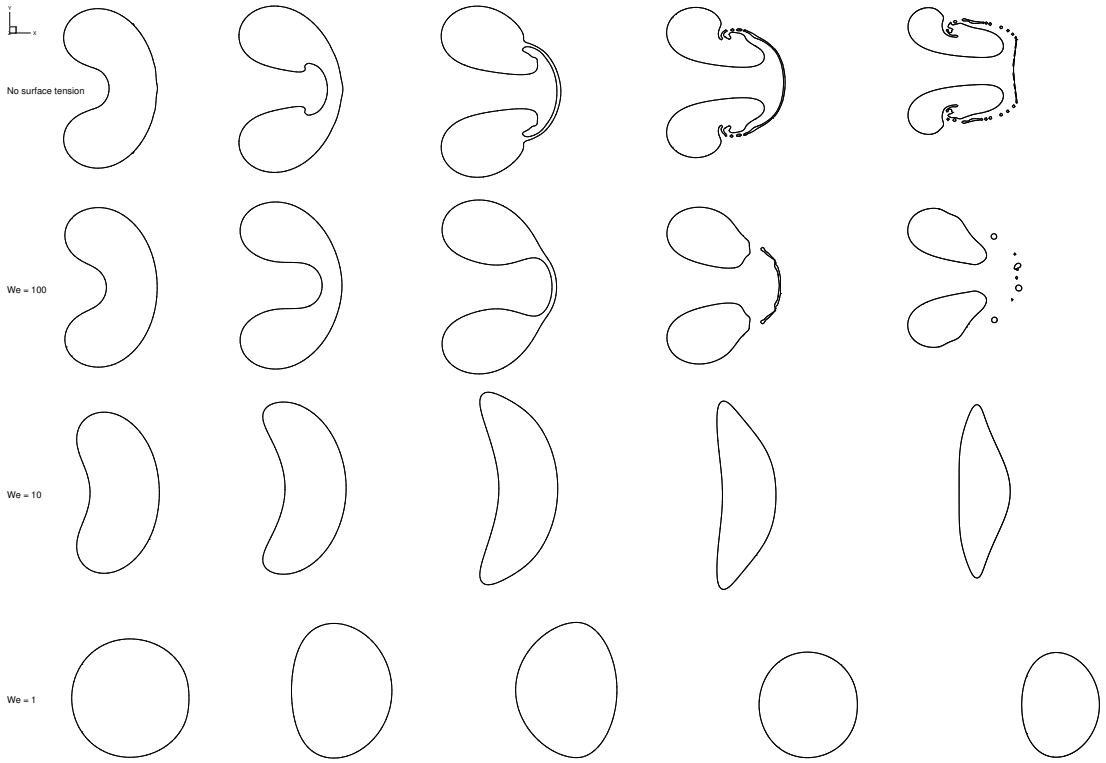
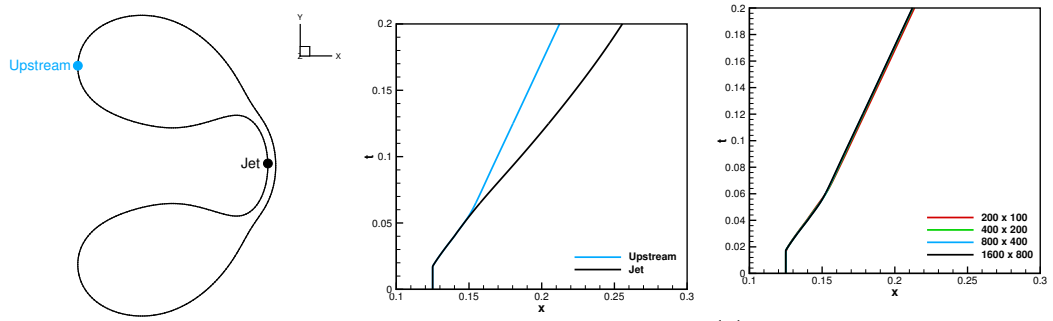


Figure 13: PLIC reconstructed interface geometries of the helium bubble at t = 0.1, 0.15, 0.2, 0.25, 0.3 (from left to right) for We = ∞, 100, 10, and 1 (from top to bottom)

Figure 13 shows a comparison of the PLIC reconstructed interface geometry for varying times and different Weber numbers. At We = 100 the formation of the side rollups of the penetrating air jet into the helium bubble is suppressed, however, the penetrating air jet still results in an ever thinner helium sheet that ruptures due to local under-resolution initially in two locations and then continues to further break up due to lack of local mesh resolution and the action of the ELVIRA normal calculation [126, 127]. Small scale bubbles are then formed due to the action of surface tension.

Increasing the surface tension to We = 10, prevents the central air jet from forming resulting in a crescent shape that deforms into an oblate shape at later times. Increasing the surface tension further to We = 1 results in oscillations of the bubble around its equilibrium circular shape. Note also that the bubble travels significantly further in the We = 1 case than in the lower surface tension cases. This is due to the fact that after shock transition, the velocity of the helium inside the bubble is larger than the velocity in the outside air. The significantly reduced cross-sectional surface area in the We = 1 case therefore results in a smaller drag force than the lower surface tension cases, leading to the bubble translating further in this case. Although the details of the interface shapes in the higher Weber number cases are markedly different from each other, their cross-sectional areas are comparable, resulting in comparable overall drag forces and thus similar positions at the same time.



(a) Interface position at $t = 0.2$

(b) Time history of interface positions

(c) Time history of upstream interface position under mesh refinement

Figure 14: Mesh refinement study of air-helium interface position with surface tension (We = 100)

Table 9: Grid Convergence Index (GCI) analysis of upstream interface position of air-helium interface at $t = 0.2$ with We $= 100$

Mesh	Interface position x	Observed order p	Richardson extrapolate x_0	GCI_{12} [%]	GCI_{23} [%]	Asymptotic range of convergence
200×100	0.2135	-	-	-	-	-
400×200	0.2125	-	-	-	-	-
800×400	0.2122	1.7245	0.2121	0.0732	0.2415	1.0013
1600×800	0.2121	1.1455	0.2120	0.0629	0.1391	1.0006

To verify that the solution converges under mesh refinement in the case with surface tension, the most upstream position of the air/helium interface was recorded as a function of time in the We $= 100$ case, see Fig. 14. Table 9 shows the result of the Grid Convergence Index (GCI) analysis following the procedure of Roache [135] for the upstream position x at $t = 0.2$. The Richardson extrapolated upstream position of the interface using the finest mesh data of 1600x800 cells is $x = 0.2120 \pm 0.0629\%$ with an asymptotic range of convergence check value of 1.0006, again very close to the theoretical value of 1, indicating the results are in the asymptotic range of grid convergence with an observed order of convergence slightly above one at 1.1455.

2.5.4.3 Shock in Air Impacting R22 Bubble

In the air shock/R22 bubble case without surface tension, the initial pre-shock conditions in the air are again set to $(\rho, u, p, \mathrm{T}, \gamma)_{air} = (1, 0, 1, 1, 1.4)$ and the conditions in the R22 are set to $(\rho, u, p, \mathrm{T}, \gamma)_{R22} = (3.117, 0, 1, 1, 1.178)$. The viscos-

ity and thermal conductivity of both air and R22 are calculated using Eq. (2.5) with $(T_0, \mu_0, S, \mathrm{Pr})_{air} = (275, 1.725 \times 10^{-5}, 110.9, 0.7)$ and $(T_0, \mu_0, S, \mathrm{Pr})_{R22} = (273, 1.15 \times 10^{-5}, 330, 0.65)$.

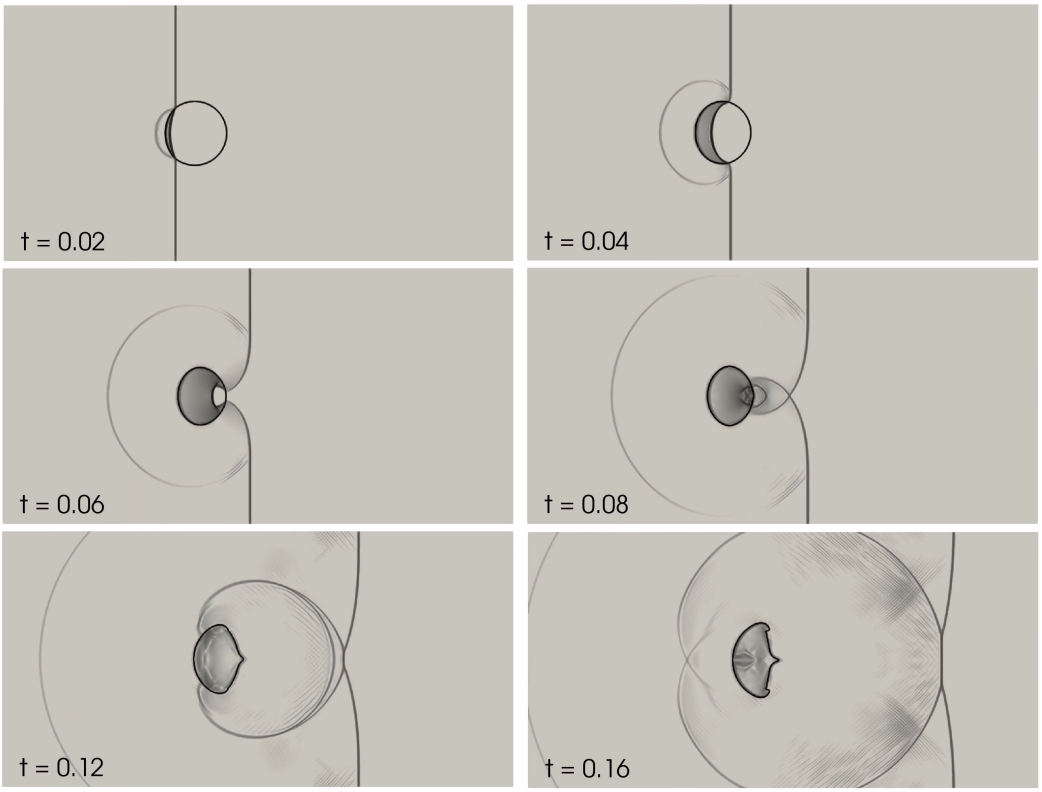


Figure 15: Numerical Schlieren images of Ma = 1.22 shock in air impacting a R22 bubble (We = ∞) at $t = 0.02$, 0.04, 0.06, 0.08, 0.12, and 0.16 time units (from top left to bottom right). Log-scale for magnitude of density gradient ranging from 5 to 4000.

Figure 15 shows numerical Schlieren images of the magnitude of the density gradient at various instances in time, and Fig. 16 shows the temperature distribution at those times. Since the ratio of specific heats for R22 is smaller than the one for

the surrounding air, the passing shock wave does not increase the temperature in the bubble as much as in the surrounding air, resulting in a slightly cooler bubble temperature as compared to the surrounding air. Furthermore, with R22 being a denser gas than air, the interaction problem is a convergent case as opposed to the previous divergent case of the helium bubble [56]. As such, the bubble deformation behavior is entirely different.

Figure 17 shows the PLIC reconstructed interface geometry at different times. Since R22 is a denser gas than air, the passing shock wave does not accelerate the R22 bubble as much as the outside air, resulting in the outside air moving faster than the R22 bubble. This results in an initial oblate shape followed by a rollup of the outside edges of the bubble. The converging refracted shocks inside the bubble result in a pressure peak in the bubble at the downstream position that creates a spike in the interface shape at the downstream position that persists at later times. The interface shape is in qualitative good agreement with the experimentally observed interface shapes [56], exhibiting both the central spike and the rollup of the outside edges.

To verify that the solution converges under mesh refinement, the position of the air/R22 interface at the upstream position was recorded as a function of time, see Fig. 18. Table 10 shows the result of the corresponding Grid Convergence Index (GCI) analysis following the procedure of Roache [135] for the interface position at

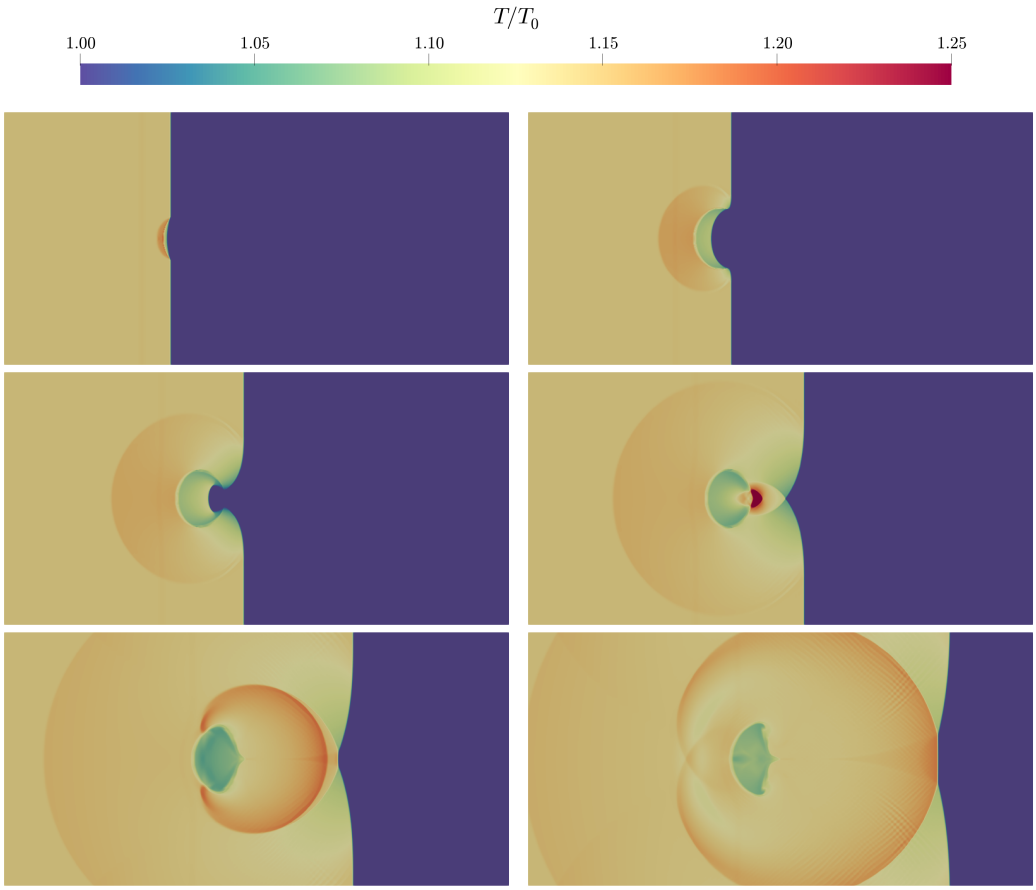


Figure 16: Temperature field normalized by pre-shock temperature of Ma = 1.22 shock in air impacting a R22 bubble (We = ∞) at $t = 0.02$, 0.04, 0.06, 0.08, 0.12, and 0.16 time units (from top left to bottom right)

$t = 0.2$. The Richardson extrapolated position of the jet using the finest mesh data of 3200×1600 cells is $x = 0.1696 \pm 0.0706\%$ with an asymptotic range of convergence check value of 1.001, very close to the theoretical value of 1, indicating the results are in the asymptotic range of grid convergence with an observed order of convergence slightly above one at 1.15.

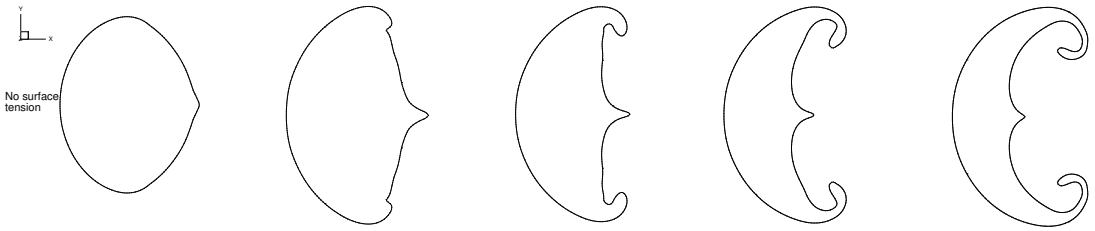
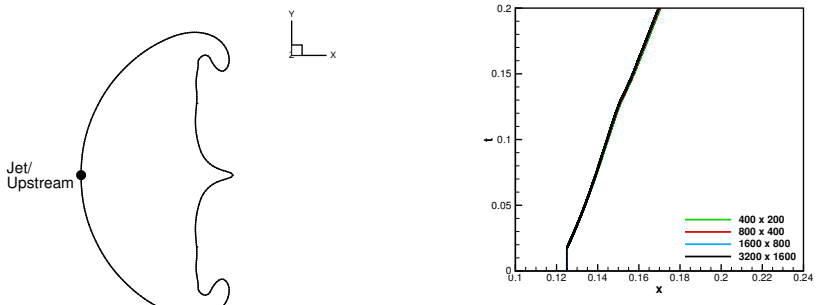


Figure 17: PLIC reconstructed interface geometries of the R22 bubble at $t = 0.1,\ 0.15,\ 0.2,\ 0.25,\ 0.3$ (from left to right)



(a) Interface position at $t = 0.2$

(b) Time history of upstream interface position under mesh refinement

Figure 18: Mesh refinement study of air-R22 interface position

Table 10: Grid Convergence Index (GCI) analysis of jet interface position of air-R22 interface at $t = 0.2$ with We $= \infty$

Mesh	Interface position x	Observed order p	Richardson extrapolate x_0	GCI_{12} [%]	GCI_{23} [%]	Asymptotic range of convergence
400×200	0.1704	-	-	-	-	-
800×400	0.1700	-	-	-	-	-
1600×800	0.1698	0.3635	0.1688	0.6788	0.8719	1.002
3200×1600	0.1696	1.1598	0.1695	0.0706	0.1576	1.001

2.5.4.4 Shock in Air Impacting R22 bubble with Surface Tension

To again demonstrate the impact of surface tension on the bubble dynamics, simulations were carried out for Weber numbers of We $= \frac{\rho u^2 D}{\sigma} = 100$, 10, and 1, in addition to We $= \infty$ of the previous section. In the definition of the Weber number, ρ and u are again the post-shock density and velocity of air, and D is the diameter of the R22 bubble. The initial and boundary conditions used are those of the prior section, however, the initial pressure in the R22 bubble is set to $p_{He} = 1 + \sigma\kappa$ to account for the presence of surface tension.

Figure 19 shows numerical Schlieren images of the magnitude of the density gradient at different instances in time for the We $= 10$ case. While the dynamics of the refracted and reflected shock wave are similar to the We $= \infty$ case, see Fig. 15, the dynamics of the interface are again markedly different. The presence of surface tension suppresses both the rollup of the sides of the bubble and the formation of the central spike that protrudes from the bubble in the We $= \infty$ case. Instead the bubble reaches an oblate shape at later times.

Figure 20 shows a comparison of the PLIC reconstructed interface geometry for varying times and different Weber numbers. At We $= 100$ the formation of the side rollups is not fully suppressed resulting in a characteristic crescent/bag shape. Increasing the surface tension to We $= 10$ prevents formation of the bag structure resulting in an oblate shape. Increasing the surface tension further to We $= 1$ results

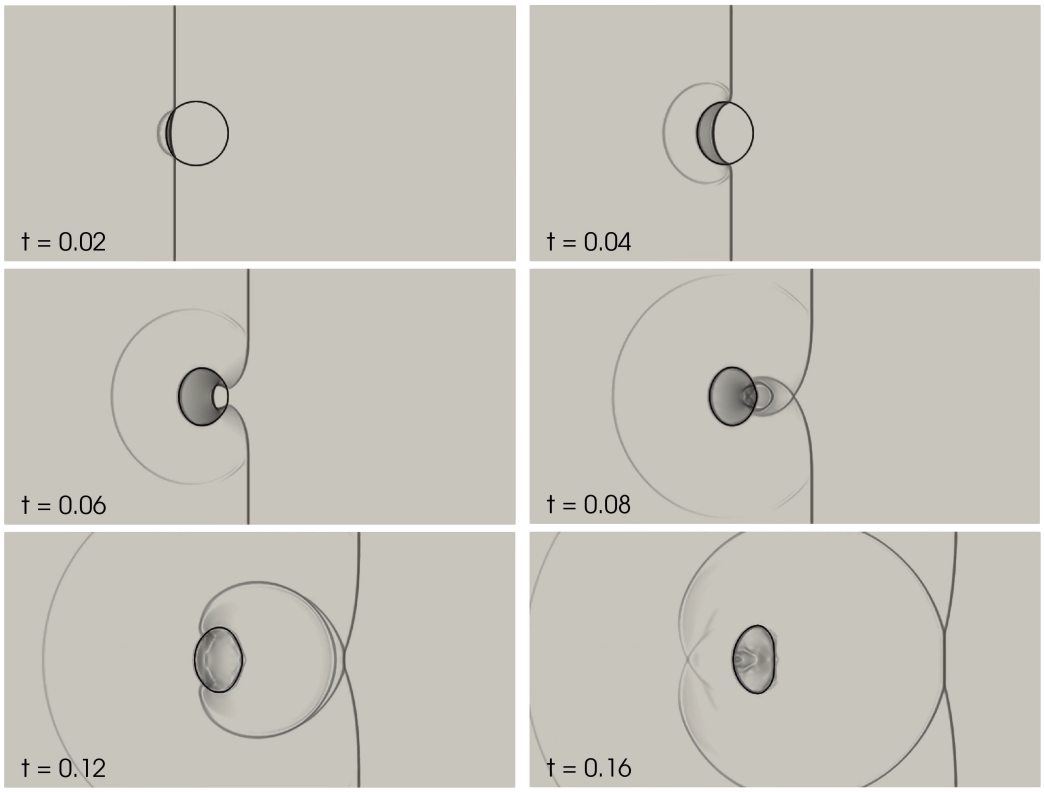


Figure 19: Numerical Schlieren images of Ma = 1.22 shock in air impacting a R22 bubble (We = 10) at various instances in time. Log-scale for magnitude of density gradient ranging from 5 to 4000.

in a barely oscillating bubble around its equilibrium circular shape. Note that unlike in the helium case, here the bubbles with We = 1 and We = 10 travel a significantly shorter distance than the lower surface tension bubbles. This again is due to the fact that the cross-sectional areas of the higher surface tension cases are smaller than the lower surface tension cases resulting in smaller drag forces. Since the outside air moves faster than the R22 bubble, this results in less aerodynamic acceleration of the bubble and therefore less displacement.

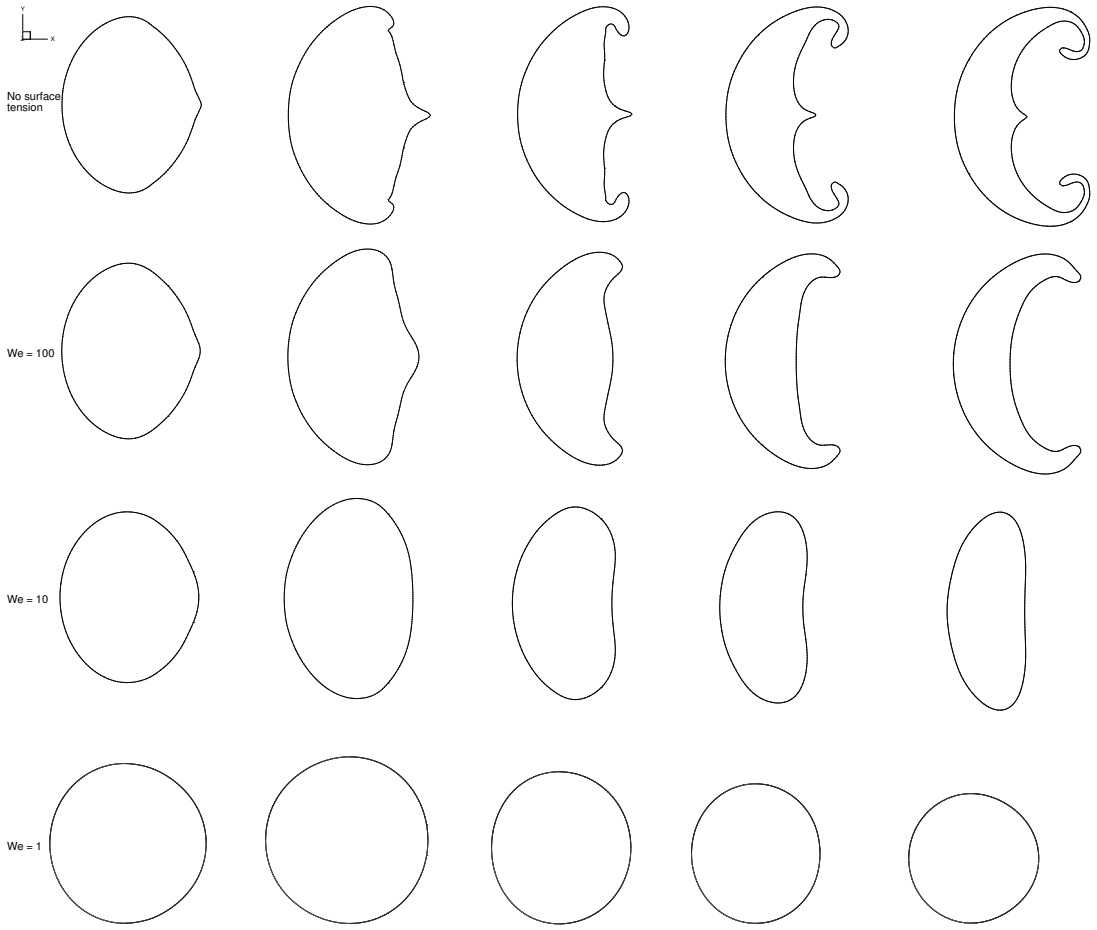


Figure 20: PLIC reconstructed interface geometries of the R22 bubble at $t = 0.1,\ 0.15,\ 0.2,\ 0.25,\ 0.3$ (from left to right) for We $= \infty$, 100, 10, and 1 (from top to bottom).

2.5.5 Three-Dimensional Shock-Bubble Interactions

To demonstrate the performance of the in-cell-reconstruction method in three dimensions, this subsection presents results for the case of shocks in air interacting with spherical bubbles of helium and R22, following the experiments of Haas and Sturtevant [56]. Again, unlike in the experiments, the fluids here are treated as immiscible and no molecular mixing occurs across the interface. In all cases, a spherical bubble of radius 0.025 is placed in the vertical and transverse center and 0.15 units from the left boundary of a rectangular three-dimensional domain of size 0.4 by 0.2 by 0.2. A planar shock of Mach number Ma = 1.22 is initialized 0.1 units from the left boundary of the domain. Simple zero-order extrapolation non-reflective boundary conditions [92] are used on all boundaries. Simulations are carried out using a mesh of $800 \times 400 \times 400$ equidistant cells. All other initial conditions are those from the corresponding two-dimensional test cases.

2.5.5.1 Shock in Air Impacting Helium Bubble

Figure 21 shows on the left the PLIC reconstructed interface between helium and air undergoing deformation due to shock interaction for the case of no surface tension (We $= \infty$). The deformation is similar to the two-dimensional case, see Fig. 13, with again a central jet of air leading to ever thinner sheets of helium in the downstream direction. These sheets again rupture due to their thickness falling below

the local mesh resolution. This, however, occurs at slightly earlier times than in the two-dimensional case. Shown in the right of Fig. 21 is the case with surface tension for We = 10. Again, the interface shapes in the three-dimensional case are similar to those in two dimensions, however the deformation occurs noticeable quicker, compare for example the interface at $t = 0.2$ in the three-dimensional case to the interface at $t = 0.3$ in the two-dimensional case.

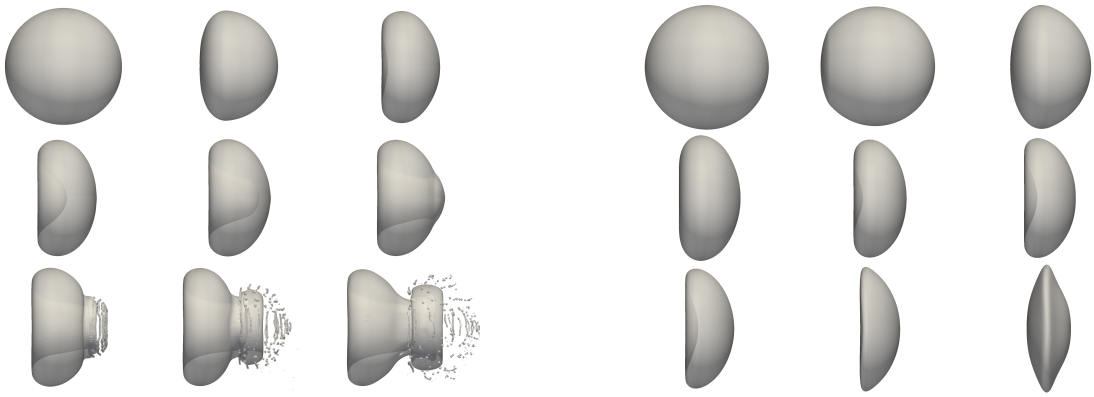


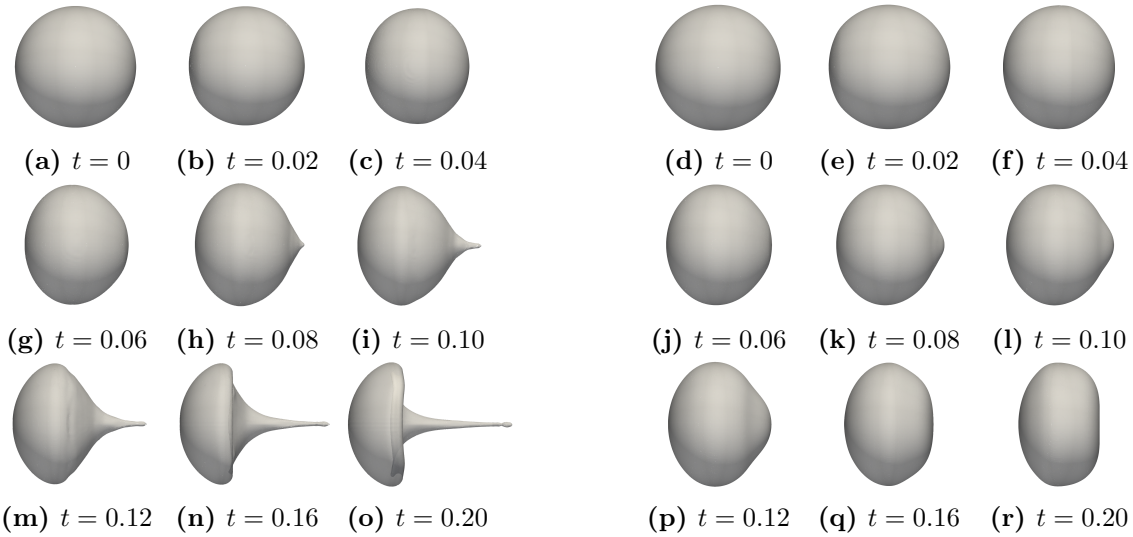
Figure 21: PLIC reconstructed interface of helium bubble deformation at $t = 0$, 0.02, 0.04, 0.06, 0.08, 0.10, 0.12, 0.16, and 0.20 (from top left to bottom right) for We $= \infty$ (left) and We = 10 (right).

Overall the interface shapes are in good qualitative agreement with the experimental observations [56], bearing again in mind that the fluids here are treated as immiscible, whereas they are miscible in the experiments.

2.5.5.2 Shock in Air Impacting R22 Bubble

The final test case consists of the shock in air interacting with a sphere of R22 in three dimensions. Figure 22 shows on the left the PLIC reconstructed interface between R22 and air undergoing deformation due to shock interaction for the case of no surface tension (We $= \infty$). The deformation is similar to the two-dimensional case, see Fig. 13, however the spike of R22 being generated in the downstream direction appears earlier and is more pronounced. This can be explained by the fact that in two dimensions, the converging shocks occur only in the one transverse direction, whereas in the three-dimensional case the refracted shock wave converges in both transverse directions resulting in a stronger pressure spike in the bubble at the downstream interface location. This behavior was also observed in the experiments [56] that show a strong spike being generated in the downstream direction that with time, however, starts to mix with the surrounding air due to the molecular mixing of the miscible fluids in the experiment. Since the fluids here are considered immiscible, the spike persists for all times.

Shown in the right of Fig. 22 is the case with surface tension for We $= 10$. Again, the interface shapes in the three-dimensional case are similar to the case in two dimensions. Again the presence of surface tension suppresses the formation of the downstream spike completely resulting in a slightly oblate shape of the bubble at later times.



(a) $t = 0$ **(b)** $t = 0.02$ **(c)** $t = 0.04$ **(d)** $t = 0$ **(e)** $t = 0.02$ **(f)** $t = 0.04$

(g) $t = 0.06$ **(h)** $t = 0.08$ **(i)** $t = 0.10$ **(j)** $t = 0.06$ **(k)** $t = 0.08$ **(l)** $t = 0.10$

(m) $t = 0.12$ **(n)** $t = 0.16$ **(o)** $t = 0.20$ **(p)** $t = 0.12$ **(q)** $t = 0.16$ **(r)** $t = 0.20$

Figure 22: PLIC reconstructed interface of R22 bubble deformation at $t = 0$, 0.02, 0.04, 0.06, 0.08, 0.10, 0.12, 0.16, and 0.20 (from top left to bottom right) for $\mathrm{We} = \infty$ (left) and $\mathrm{We} = 10$ (right).

2.6 Conclusions

A hybrid capturing/tracking method for compressible multi-fluid flows that couples an un-split, geometric volume tracking method with a finite volume wave propagation scheme is presented. The method maintains the interface between the immiscible fluids as sharp by directly enforcing the jump conditions across the interface.

For cells containing both fluids and therefore an interface, individual states on either side of the interface are reconstructed such that cell-face Riemann problems can be solved within each fluid separately. Instead of updating the individual reconstructed fluid states in mixed cells, cell face area fraction weighted averages of the single fluid Riemann problem waves are computed to update the mixed state of cells containing

the interface directly. The approach is used to simulate a number of test cases in one, two, and three dimensions, obtaining good agreement with theoretical results and experimental observations. While the presented method works well for fluids with similar impedance and is limited to fluids that can be described by equations of state for ideal gases, the inherent assumption of linear wave interaction of cell face waves with the fluid interface breaks down for fluids with vastly different impedances, like liquid/gas systems that would be described by general equations of state. In these cases, the interaction of face waves with the then phase interface has to be taken explicitly into account and the reconstruction equations have to be modified to account for general equations of state. Such an approach will be explored in future work.

To couple the un-split volume of fluid interface capturing scheme to the in-cell reconstruction method presented in this paper an extension of the method originally proposed by Owkes and Desjardins [119] is presented based on the generation of triangular prism flux volumes. This extension also provides a path to apply the geometric un-split volume of fluid transport scheme to arbitrary polyhedral unstructured meshes, an approach that will be explored in future work.

Finally, it should be pointed out that the presented method can in principle be applied to other types of interfaces as well, for example deflagration waves, detonation waves, or shocks. All that is required is a modification of the jump conditions across the interface and the inclusion of relative motion between the interface and the background flow if such motion exists.

Chapter 3

EXTENSION OF IN-CELL RECONSTRUCTION TO INCLUDE GENERAL EQUATION OF STATE

In this chapter, an extension to the novel numerical method using in-cell reconstruction described in Chapter 2 to allow for a more general equation of state is outlined. First, the thermodynamic changes involved with using a more general equation of state and its consequences are described. Beyond that, details of numerical methods needed to incorporate into the existing model are described. Finally, the outcome of the extension efforts are summarized.

3.1 Thermodynamic Considerations

The governing equations used in the proposed method in Chapter 2 use the equation of state for an ideal gas Eq. (2.6), which is not suitable for flows involving high impedance such as liquids. For the target application of primary atomization of a liquid in supersonic flows, it is crucial that the method is extended to accommodate liquids. Therefore, a more general equation of state and more specifically, a jump in equation of state is required. For the purposes of this research work, it has been identified that the stiffened equation of state of [57] is appropriate,

$$p = (\gamma - 1)\ \rho i\ - \gamma p_{\infty} \tag{3.1}$$

where, p_∞ is a modeling parameter specific to the fluid of interest [24]. This change results in the speed of sound c being,

$$c = \sqrt{\gamma \frac{p + p_\infty}{\rho}} \tag{3.2}$$

and the temperature T being,

$$T = \frac{p + p_\infty}{(\gamma - 1)\ c_v\, \rho}\,, \tag{3.3}$$

Furthermore, a direct consequence is the change to the equations used to reconstruct the primitive variables from the cell-averaged variable for a mixed cell (see Eqs. (2.24) to (2.26)). The new equations due to the introduction of Eq. (3.1) then are,

$$\begin{aligned}
u_0 = u_1 &= \frac{\overline{\rho u}}{\overline{\rho}}, \quad v_0 = v_1 = \frac{\overline{\rho v}}{\overline{\rho}}, \quad w_0 = w_1 = \frac{\overline{\rho w}}{\overline{\rho}}, \\
p_0 &= \frac{(\gamma_0 - 1)\,(\gamma_1 - 1)\left(\overline{\rho E} - \frac{1}{2}\rho\ ||\boldsymbol{u}||^2\right)}{\alpha\,(\gamma_0 - 1) + (1 - \alpha)\,(\gamma_1 - 1)} - \frac{\alpha\,(\gamma_0 - 1)\left(\sigma\,\kappa + [\![\boldsymbol{n}^T \cdot \overline{\overline{\tau}} \cdot \boldsymbol{n}]\!]_\Gamma + \gamma_1\, p_{\infty,1}\right)}{\alpha\,(\gamma_0 - 1) + (1 - \alpha)\,(\gamma_1 - 1)} \\
&\quad - \frac{(1 - \alpha)\,(\gamma_1 - 1)\,\gamma_0\, p_{\infty,0}}{\alpha\,(\gamma_0 - 1) + (1 - \alpha)\,(\gamma_1 - 1)}, \\
p_1 &= p_0 + \sigma\,\kappa + [\![\boldsymbol{n}^T \cdot \overline{\overline{\tau}} \cdot \boldsymbol{n}]\!]_\Gamma, \\
\rho_0 &= \frac{\overline{\rho}\ (p_0 + p_{\infty,0})\,(\gamma_1 - 1)\,c_{v,1}}{\alpha\ (p_1 + p_{\infty,1})\,(\gamma_0 - 1)\,c_{v,0} + (1 - \alpha)\ (p_0 + p_{\infty,0})\,(\gamma_1 - 1)\,c_{v,1}}, \\
\rho_1 &= \frac{\overline{\rho}\ (p_1 + p_{\infty,1})\,(\gamma_0 - 1)\,c_{v,0}}{\alpha\ (p_1 + p_{\infty,1})\,(\gamma_0 - 1)\,c_{v,0} + (1 - \alpha)\ (p_0 + p_{\infty,0})\,(\gamma_1 - 1)\,c_{v,1}}
\end{aligned} \tag{3.4}$$

As before, due to the dependence of viscosity on temperature (Eq. (2.5)), there arises a non-linear equation in temperature, which is the same as before (Eq. (2.25)). However, the shorthand coefficients A_0 through A_3 in the non-linear equation are

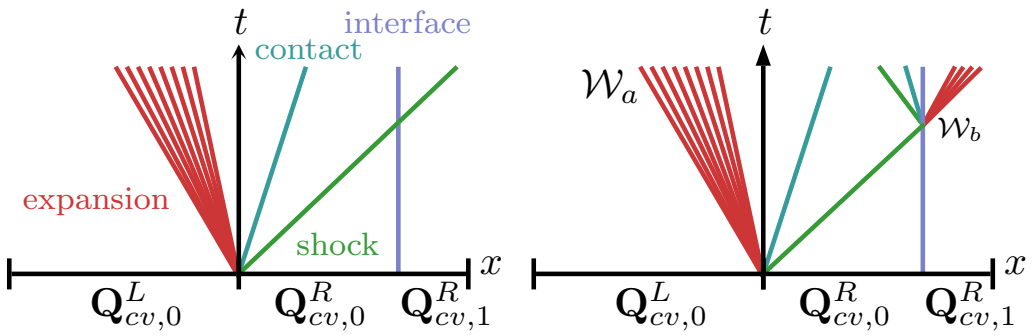
now given by,

$$
\begin{aligned}
A_0 &= \bar{\rho}\, c_{v,0}\, c_{v,1}\, (\gamma_0 - 1)\, (\gamma_1 - 1) \\
A_1 &= \frac{\alpha\, (1-\alpha)\, (\gamma_0 - 1)\, (\gamma_1 - 1)\, (c_{v,1} - c_{v,0})}{\alpha\, (\gamma_0 - 1) + (1-\alpha)\, (\gamma_1 - 1)} \\
A_2 &= \alpha\, (\gamma_0 - 1)\, c_{v,0}\, (\sigma\kappa + p_{\infty,1}) + (1-\alpha)\, (\gamma_1 - 1)\, c_{v,1}\, p_{\infty,0} \\
A_3 &= \left[\frac{(\gamma_0 - 1)\, (\gamma_1 - 1)\, \left(\overline{\rho E} - \frac{1}{2}\rho\, ||\boldsymbol{u}||^2\right)}{\alpha\, (\gamma_0 - 1) + (1-\alpha)\, (\gamma_1 - 1)} - \frac{\alpha\, (\gamma_0 - 1)\, (\sigma\kappa + \gamma_1\, p_{\infty,1})}{\alpha\, (\gamma_0 - 1) + (1-\alpha)\, (\gamma_1 - 1)} \right. \\
&\quad \left. - \frac{(1-\alpha)\, (\gamma_1 - 1)\, \gamma_0\, p_{\infty,0}}{\alpha\, (\gamma_0 - 1) + (1-\alpha)\, (\gamma_1 - 1)} \right] (\alpha\, (\gamma_0 - 1)\, c_{v,0} + (1-\alpha)\, (\gamma_1 - 1)\, c_{v,1})
\end{aligned}
\tag{3.5}
$$

3.2 Numerical Methods

The core of the numerical approach described in Chapter 2 involves reconstructing individual fluid properties from the cell-averaged values locally in a cell using Eqs. (2.24) to (2.25), and solving Riemann problems using the wave-propagation method, see Section 2.4. The wave contribution from each phase is then added to the cell-average weighted by the liquid face-area fraction β, which is calculated as the average of the left and right cells' values (see Section 2.4.2.2). As discussed briefly in Section 2.4.2.2, this is a linearization and numerical tests have shown that this linearization is viable only if the fluids considered are of comparable impedance ($p_\infty = 0$), such as ideal gases. This is due to the inherent assumption that the single-phase waves originating from cell faces do not further interact with the phase

interface itself within a single time-step, see Fig. 23a for an example. For liquid/gas systems, the interaction of waves originating from the cell-face Riemann problems with the phase interface has to be taken into account [94, 154].

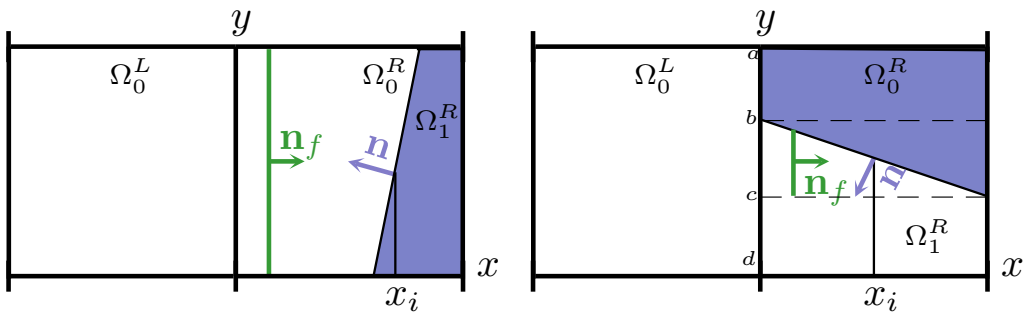


(a) Linearized wave-interface interaction **(b)** Full wave-interface interaction

Figure 23: Interaction of waves from cell-face Riemann problem with phase interface in $xt-$plane

For this purpose an exact two-phase Riemann solver has been developed following the procedure by Kamm [77] that allows for an arbitrary jump in the equation of state. If within a time step it is detected that a wave from the single-phase cell-face Riemann problems passes over the interface, a new two-phase Riemann problem at the phase interface is solved. This detection is achieved by calculating the interaction time based on the wave speeds of the cell-face Riemann problem and the phase interface position in the cell. For a PLIC-based phase interface reconstruction [140], the incoming wave interacts with the phase interface at different positions throughout the duration of a time-step. For the purposes of estimating the interaction time, it is proposed to use the mid-point location between the interface location first passed

by the face wave and the smaller of the interface location last passed by the wave and the wave location at the end of the time step, see for example x_i in Fig. 24a. If the cell face wave is found to interact with the phase interface, a two-phase Riemann problem is set up in the frame of reference of the phase interface.



(a) Representative of Fig. 23 **(b)** Mixed-phase cell face

Figure 24: In-cell wave-interface PLIC-based interaction in physical $xy-$plane

For example, as shown in Fig. 23b, the fluid-0 phase Riemann problem solved at the cell face gives rise to a wave system $\mathcal{W}_a^p$ consisting of shocks, contact waves and rarefactions. At the interface location x_i, the left state of the two-phase Riemann problem is $\boldsymbol{Q}_0^R - \mathcal{W}_{a,0}^3$ rotated into the interface normal $\boldsymbol{n}$ direction, and the right state is $\boldsymbol{Q}_{cv,1}^R$ rotated into the interface normal direction as well. In the example shown in in Fig. 23, the $p = 3$ wave interacting with the phase interface is a shock. The interface two-phase Riemann problem is then solved using an exact two-phase Riemann solver, resulting in a wave system $\mathcal{W}_b^p$. The resulting waves and wave contributions are then rotated back into the frame of reference of the cell face $\boldsymbol{n}_f$.The fluctuations and cell

volume averages of $\boldsymbol{Q}_{cv}$ are then updated using these waves. Note that the cell face tangential components of the resulting wave system $\mathcal{W}_b^p$ can affect mesh cells in the transverse direction and will likely have to be considered separately.

In general, a cell face is further separated into regions of single-phase Riemann problems with and without phase interface interactions. For example, consider the mixed-phase cell face shown in Fig. 24b. The cell face is split into regions ab, bc and cd. Cell face waves originating from regions ab and cd stay within their respective phases for the duration of the time-step and cannot interact with the phase interface. Cell face waves originating from the cell face region bc, however, can interact with the phase interface necessitating the interaction procedure outlined above.

3.3 Verification Results

In this section, various test cases are performed ranging from one- to two-dimensional cases to verify the extension described above is able to handle systems of gas/liquid flows.

3.3.1 1-D Shock in Air Impacting an Air-Water Interface

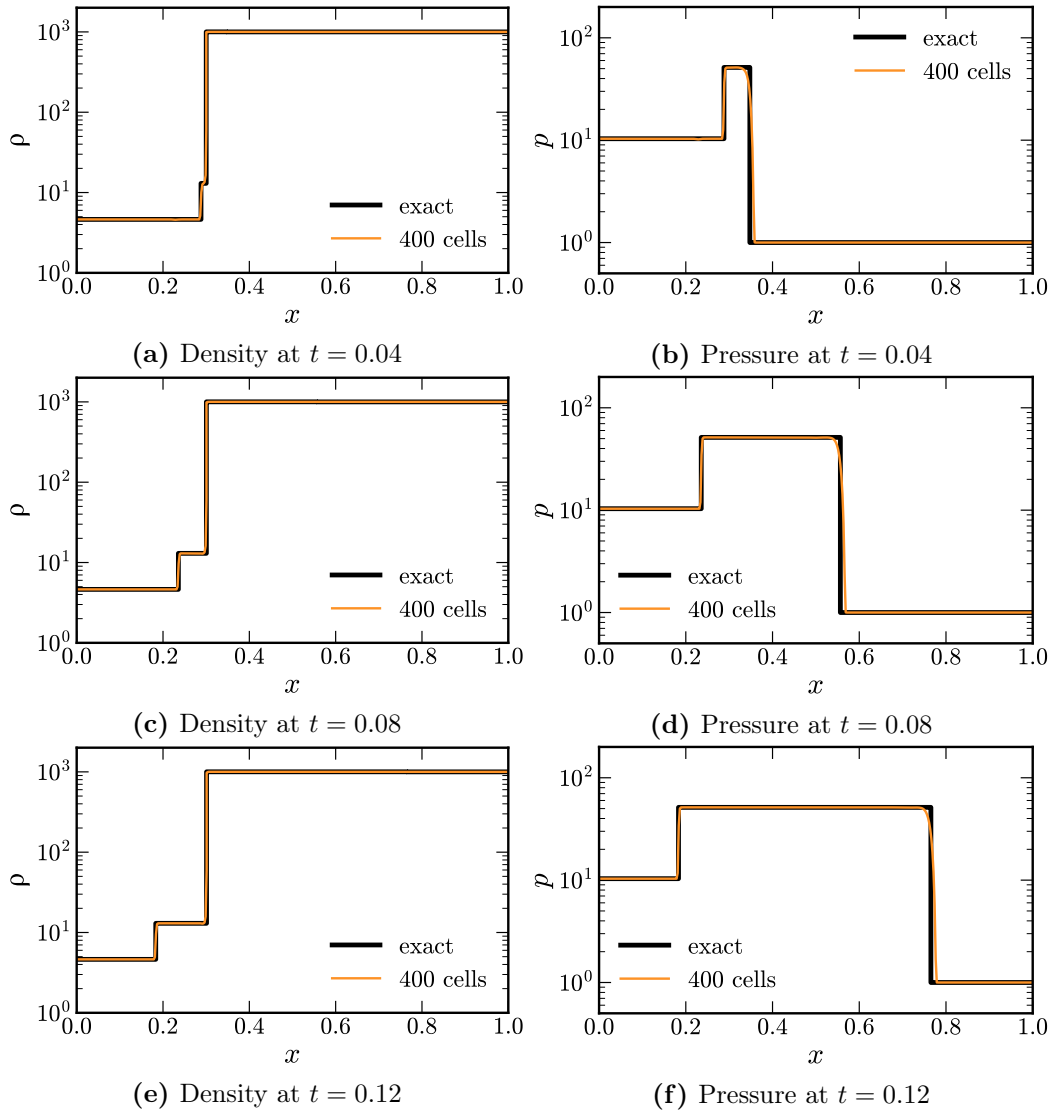
The first verification problem consists of a Ma = 3 shock in air impacting an air/water phase interface. The shock is placed initially at $x = 0.2$ inside a unit sized domain, with the air/water phase interface located at $x = 0.3$. The initial conditions

in the air are $(\rho, u, p, \gamma, p_\infty, \mu, \mathrm{Pr})_0 = (1.2, 0, 1, 1.4, 0, 1.813 \times 10^{-5}, 0.7)$, and in the water $(\rho, u, p, \gamma, p_\infty, \mu, \mathrm{Pr})_1 = (1000, 0, 1, 5.5, 4921.15, 1.002 \times 10^{-3}, 7)$. A mesh of 400 equidistant cells is used with CFL $= 0.5$. Figure 25 compares the computed density and pressure solution with the exact solution at different instances in time. It should be noted that the exact solution is from an exact Riemann solver for the compressible Euler equations while the computed solution solves the full compressible Navier-Stokes equations.

The computed solution is in good agreement with the exact solution, which consists of both reflected and transmitted shocks. The phase interface remains a discontinuity without any visible pressure oscillations. Although the transmitted shock in the liquid exhibits a strong pressure jump, the associated density jump is practically invisible in the density solution due to the high stiffness in the equation of state for water.

3.3.2 1-D Shock in Water Impacting a Water-Air Interface

The second verification problem consists of a Ma $= 1.01$ shock in water impacting a water/air phase interface. The shock is initially placed at $x = 0.6$ inside a unit sized domain, with the water/air phase interface located at $x = 0.7$. The initial conditions in the water are $(\rho, u, p, \gamma, p_\infty, \mu, \mathrm{Pr})_1 = (1000, 0, 1, 5.5, 4921.15, 1.002 \times 10^{-3}, 7)$ and in the air $(\rho, u, p, \gamma, p_\infty, \mu, \mathrm{Pr})_0 = (1.2, 0, 1, 1.4, 0, 1.813 \times 10^{-5}, 0.7)$. A mesh of 400 equidistant cells is used with a CFL number of 0.5.



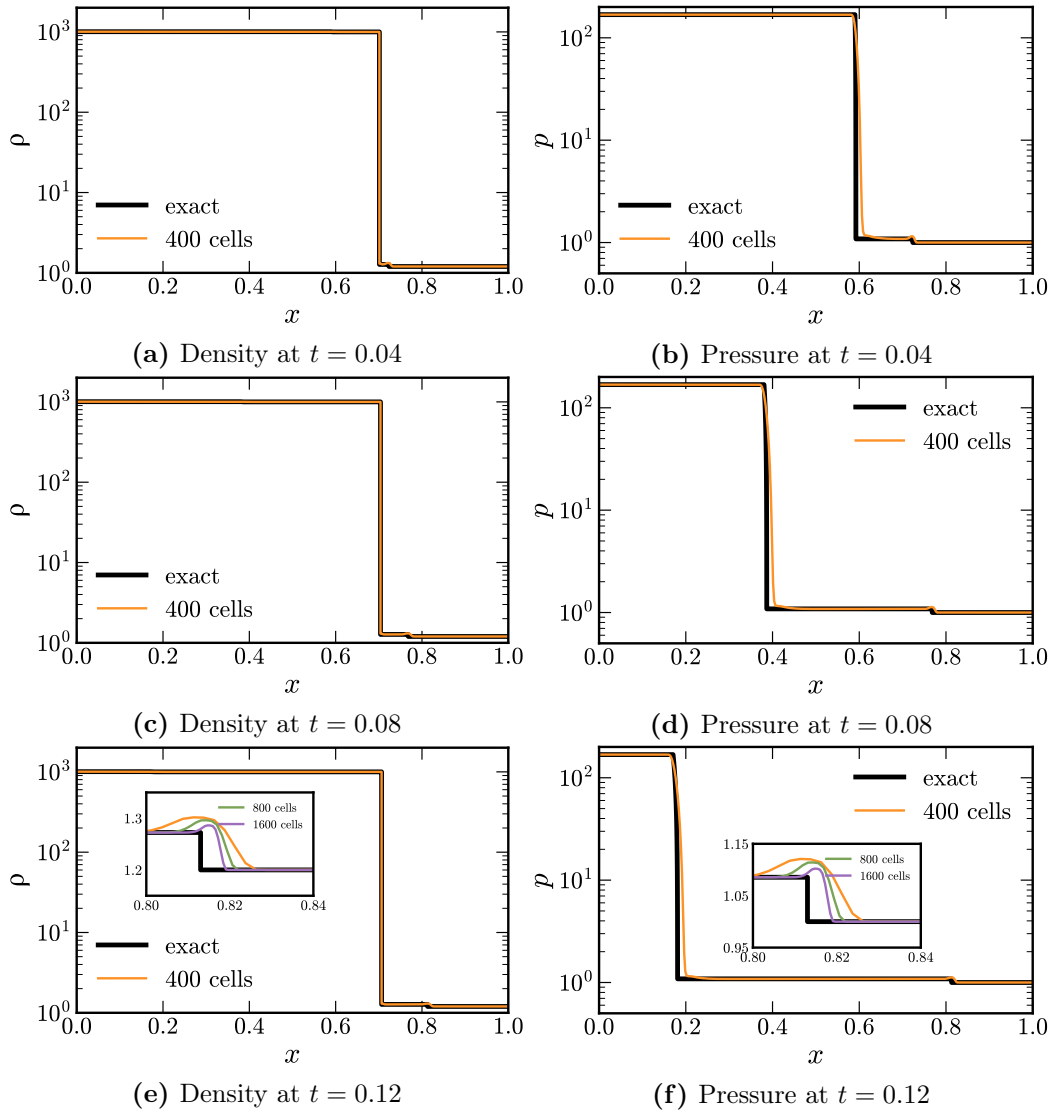
(a) Density at $t = 0.04$ **(b)** Pressure at $t = 0.04$

(c) Density at $t = 0.08$ **(d)** Pressure at $t = 0.08$

(e) Density at $t = 0.12$ **(f)** Pressure at $t = 0.12$

Figure 25: Solutions for Ma = 3 shock in air impacting an air-water interface using log-scale

Figure 26 compares the computed density and pressure solution with the exact solution at different instances in time. The computed solution is in very good agreement with the exact solution that consists of a reflected rarefaction wave in the



(a) Density at $t = 0.04$ **(b)** Pressure at $t = 0.04$

(c) Density at $t = 0.08$ **(d)** Pressure at $t = 0.08$

(e) Density at $t = 0.12$ **(f)** Pressure at $t = 0.12$

Figure 26: Solutions for Ma = 1.01 shock in water impacting a water-air interface using log-scale

liquid and a transmitted shock in the gas. The phase interface remains a discontinuity

without any visible pressure oscillations. On close inspection, the transmitted shock in the gas exhibits a small overshoot after the shock that however decreases with mesh refinement (see Figs. 26e and 26f).

3.3.3 2-D Shock in Air Impacting an Air-Water Interface

The final verification problem consists of a Ma = 1.3 shock in air impacting a two-dimensional water drop of diameter 0.05 with center at $x = 0.15$, following the experimental work by [70]. The computational domain is a 0.4×0.2 rectangular box. The initial conditions in the air are $(\rho, u, p, \gamma, p_\infty, \mu, \mathrm{Pr})_0 = (1.2, 0, 1, 1.4, 0, 1.813 \times 10^{-5}, 0.7)$, and in the water are $(\rho, u, p, \gamma, p_\infty, \mu, \mathrm{Pr})_1 = (1000, 0, 1, 5.5, 4921.15, 1.002 \times 10^{-3}, 7)$. Surface tension is neglected in this case. The shock is placed initially at $x = 0.1$ units from the left boundary. Simulations are carried out for a coarse mesh of 200×100 equidistant cells.

Figure 27 shows numerical Schlieren images of the simulation results at various instances in times. The actual sharp phase interface location is indicated by a thin red line. As shown, the shock wave passing over the drop is partially reflected and transmitted very weakly through the drop. The drop starts to move and, due to the lack of surface tension, begins to deform. Unfortunately, the simulation fails for longer times or with mesh refinement. This is a consequence of using only the normal contributions of the two-phase Riemann problem solved at the phase interface, ignoring the transverse contributions as described in Section 3.2.

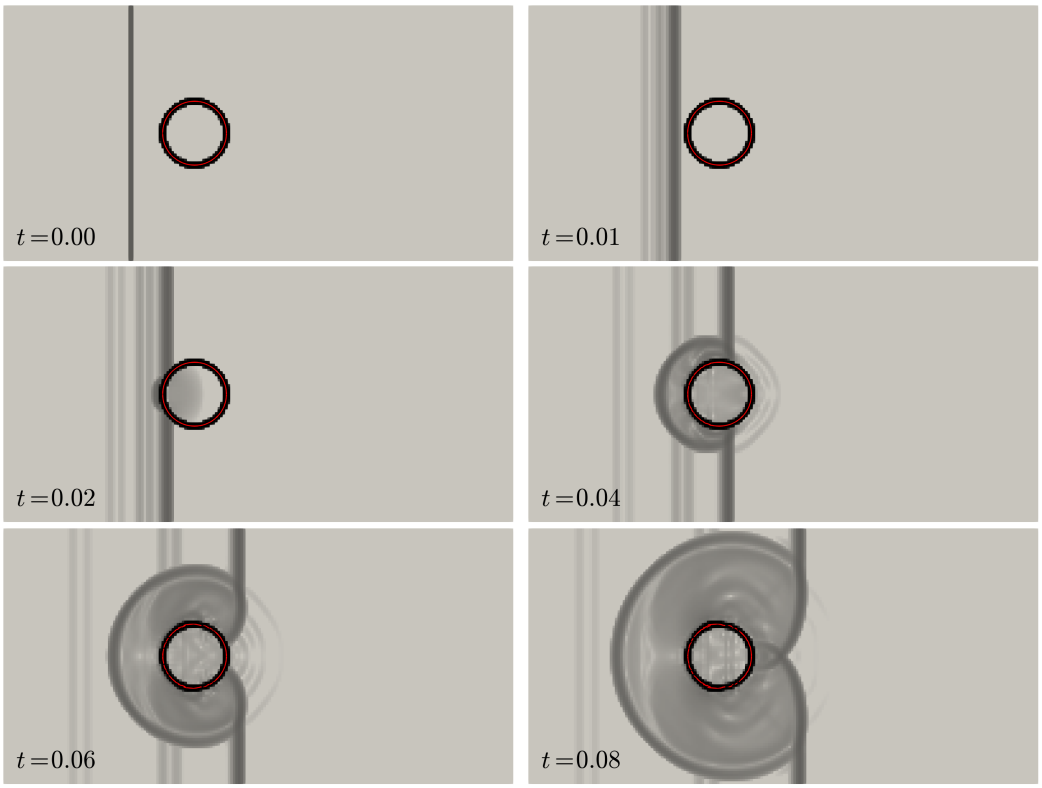


Figure 27: Numerical Schlieren images of a Ma = 1.3 shock in air impacting a water drop at various instances in time. Colored by log-scale for magnitude of density gradient ranging from 0.1 to 100,000. PLIC-reconstructed phase interface colored by red line.

3.4 Conclusions

A direct extension to the wave-propagation method outlined in Section 2.4 is proposed, wherein the interaction of waves emanating outwards from cell-face Riemann problems with phase interfaces are considered. Additional two-phase Riemann problems are set up and solved at the phase interfaces to generate new set of waves which further affect the solution. Since the phase interface is arbitrarily oriented in

space and the cell-faces are aligned with the coordinate axes, the new set of waves consist of both normal and tangential contributions to the solution. For the sake of simplicity, only normal contributions are included and a range of test cases are performed. One-dimensional test cases show very good agreement with analytical solutions, whereas multi-dimensional test cases fail immediately or eventually in time due to the missing tangential contributions. The addition of tangential contributions involves considerable complexity to the existing algorithm and regardless, the robustness of this approach for gas/liquid flows in multi-dimensional cases remains unclear. Furthermore, the topic of increasing the formal order of accuracy in the vicinity of the phase interface using this approach is non-trivial. In the following chapter, a new approach is considered using a diffuse-interface method in the quest for addressing the aforementioned concerns.

Chapter 4

A DIFFUSE-INTERFACE APPROACH TO SIMULATING COMPRESSIBLE MULTICOMPONENT FLOWS WITHIN AN ADAPTIVE MESH REFINEMENT FRAMEWORK

Karthik Kannan[3], Fabian Fritz [4], Nico Fleischmann[4], Carlos Ballesteros[3], Stefan Adami [5], Nikolaus A. Adams[5], Marcus Herrmann[6]

A paper to be submitted to the Journal of Computational Physics

4.1 Abstract

A diffused-interface method for solving the compressible, multicomponent Navier-Stokes equations, i.e. the quasi-conservative five-equation model (Allaire, Clerc, and Kokh [3]) including capillary and viscous effects (Coralic and Colonius [28] and Garrick, Hagen, and Regele [45]) is extended to be used in conjunction with a novel, unstructured, cell-based adaptive mesh refinement (AMR) library (Ballesteros et al. [7]). Low dissipation, high spatial accuracy is obtained by using a weighted essentially

[3]Ph.D. student, Arizona State University

[4]Ph.D. student, Technical University of Munich

[5]Faculty, Technical University of Munich

[6]Associate Professor, Arizona State University

non-oscillatory (WENO-Z) (Borges et al. [16]) discretization, while the numerical smearing of the material interface is controlled using a tangent of hyperbola for interface capturing (THINC) scheme (Shyue and Xiao [155]). This ensures higher-order while limiting the material interface width to 2-3 mesh cells. Large stencils required for the use of higher-order schemes in the unstructured framework are generated consistent with AMR refinement and coarsening operations. The standard height function method (Cummins, Francois, and Kothe [29]) for estimating interface curvature is extended to account for the increased thickness of the material interface. The use of AMR helps achieve higher mesh resolution in regions of liquid and shock discontinuities, which is crucial to compressible atomization applications. Simulation results for test cases that involve shock-interface interaction are used to demonstrate the ability to resolve desired flow features while maintaining the performance of the higher-order methods used.

4.2 Introduction

Compressible fluid flows with multiple physical states of matter occur in many engineering applications such as shock-induced mixing of fuel and oxidizer in supersonic combustion [176, 4, 87], cavitation-induced bubble collapse in naval applications [37, 84, 21], underwater explosions [171, 170], Richtmyer-Meshkov type instabilities in inertial confinement fusion [53, 178], and in medical applications such as shock wave lithotripsy [125, 75] to treat kidney stones. The challenges involving the study of these

physical processes arise due to complex interactions between shock waves and phase interfaces that make it difficult to perform detailed analysis through experiments. Moreover, the length and time scales that occur in some of these processes create additional technical challenges in measuring quantities of interest that might aid in design of engineering systems that use these processes as an underlying mechanism [69, 67, 87]. Therefore, developing robust numerical schemes that can capture these physical processes accurately are of prime interest to aid with experimental analysis.

Many numerical approaches to modeling compressible multiphase flows have been developed, and they can be broadly classified on how the phase interface between the fluids is treated. A method that maintains the phase interface as a discontinuity, with the aim to replicate the physical nature is called a sharp interface method (SIM), whereas a method that allows the phase interface to spread over a certain thickness is called a diffuse interface method (DIM). As each approach has its own benefits and shortcomings, there has been considerable research towards creating robust numerical methods using both approaches and are briefly discussed subsequently.

Generally, SIM approaches maintain and transport a discontinuous interface description either by means of level-set methods [43, 83, 118, 76, 39, 20], geometric volume-of-fluid (VoF) methods [115, 14, 78], front-tracking methods [49, 50] or arbitrary Lagrangian-Eulerian (ALE) methods [105, 167, 32, 41]. One of the main challenges in these approaches is avoiding dissipative errors across the discontinuous fluid interface, with solutions ranging from using ghost-fluid methods to extrapolate values thereby eliminating a stencil across the interface [43, 65, 64], to solving two-

phase Riemann problems at the phase interface identified by geometric means [26, 39], to in-cell reconstruction of individual phases from cell averages [78]. Furthermore, as the interface is usually captured separately from the compressible flow equations, a correct coupling of these systems must enforce consistency between the transport of volume and transport of density through appropriate thermodynamic treatment at the interfaces. Finally, the complexity of implementing SIM approaches, especially in higher dimensions, can require extensive geometric considerations.

On the other hand, DIM approaches are relatively easier to implement, even in higher dimensions, due to the nature of imposing a smooth transition between phases. One of the main advantages of this approach is the use of a single system of equations which allows consistent numerical treatment of the entire two-phase flow description. The biggest challenge in this approach is restricting the diffusion of the interface region, which can grow uninhibited without any special treatment. Early efforts in this regard involved using higher-order reconstruction schemes [74, 28] such as the weighted essentially non-oscillatory (WENO) scheme [148]. However, over long periods of time, the interface region was still shown to increase. Later efforts sought ways of re-sharpening the diffuse interface. So, Hu, and Adams [158] introduced an anti-diffusion correction approach which was performed as post-processing to the volume fraction field. An interface compression technique was proposed by Shukla, Pantano, and Freund [150] which introduced non-conservative operators and also relied on a post time-step sharpening step. A later extension by Tiwari, Freund, and Pantano [164] brought about the same interface compression within the solution

time-step by means of modifying the governing equations. Alternatively, the tangent of hyperbola for interface capturing (THINC) scheme of Xiao, Ii, and Chen [175] and Shyue and Xiao [155] uses a hyperbolic tangent function to reconstruct the variables without needing modifications to the governing equations, or the introduction of an additional post-processing step. The THINC schemes have later been also used successfully by Garrick, Hagen, and Regele [45]. Furthermore, Chiapolino, Saurel, and Nkonga [23] introduced a flux limiter to sharpen the diffuse interface that was applicable only to interfaces and thus requires a robust interface indicator function. Finally, a family of boundary-variation-diminishing (BVD) schemes [33] have been used for the inviscid two-phase compressible flows.

Due to the relative simplicity of implementation of DIM approaches for higher dimensions, they make a good candidate for use in conjunction with an adaptive mesh refinement (AMR) method of providing high resolution in regions of interest. In the context of compressible multiphase flows, Pau et al. [122] used an unsplit second-order Godunov method with a block-based AMR to simulate compressible two-phase porous flow. Majidi and Afshari [107] used a THINC and a second-order monotonic upstream-centered scheme for conservation laws (MUSCL) scheme with a block-based AMR to simulate compressible gas-liquid flows. More recently, Schmidmayer et al. [142] used a second-order MUSCL scheme with a tree-based AMR approach. To the author's best knowledge, higher-order methods with AMR for a DIM approach in the context of compressible multiphase flows has not been studied yet.

The purpose of the present contribution is to provide a higher-fidelity approach to simulating compressible multiphase flows using existing DIM approaches by means of a hybrid reconstruction method. The approach proposed here utilizes a low-dissipation, high-order WENO-Z scheme [16] in regions of pure phase thereby providing high spatial accuracy, and switches to a THINC scheme [175, 155] near the vicinity of the interface to avoid further dissipation. High-resolution in regions of interest is provided in regions of interest, using a novel cell-based unstructured Cartesian AMR framework [7] which allows for very high resolution with an overall lower computational expense. Furthermore, an extension to the standard height functions [29] is proposed for a DIM approach, which in conjunction with a redistribution scheme provides estimation of interface curvature in the diffuse region while also being mesh convergent.

The rest of this paper is structured as follows: Section 4.3 describes the governing equations, model closures and mixture rules required to provide a complete flow description of compressible two phase flows, Section 4.4 describes the various numerical methods used to solve the governing equations, Section 4.5 describes the various tests used to verify the methods used and its implementation, and finally Section 4.6 summarizes the present work and outlines potential areas of future work.

4.3 Governing Equations

4.3.1 Five-Equation Model

The equations governing the evolution of compressible multicomponent flows is described by the five-equation model of Allaire, Clerc, and Kokh [3] including capillary and viscous effects [123, 46],

$$\frac{\partial \rho_1 \alpha_1}{\partial t} + \nabla \cdot (\rho_1 \alpha_1 \boldsymbol{u}) = 0 \tag{4.1a}$$

$$\frac{\partial \rho_2 \alpha_2}{\partial t} + \nabla \cdot (\rho_2 \alpha_2 \boldsymbol{u}) = 0 \tag{4.1b}$$

$$\frac{\partial \rho \boldsymbol{u}}{\partial t} + \nabla \cdot \left(\rho \boldsymbol{u}\boldsymbol{u} + p\overline{\overline{I}} \right) = \nabla \cdot \overline{\overline{\tau}} + \sigma \kappa \nabla \alpha_1 \tag{4.1c}$$

$$\frac{\partial \rho E}{\partial t} + \nabla \cdot \left((\rho E + p)\, \boldsymbol{u} \right) = \nabla \cdot \left(\overline{\overline{\tau}} \cdot \boldsymbol{u} \right) + \sigma \kappa \nabla \alpha_1 \cdot \boldsymbol{u} \tag{4.1d}$$

$$\frac{\partial \alpha_1}{\partial t} + \boldsymbol{u} \cdot \nabla \alpha_1 = 0 \tag{4.1e}$$

where $\rho_1\alpha_1$, $\rho_2\alpha_2$ correspond to the liquid and gas phasic densities, α_1 and α_2 the liquid and gas volume fractions, ρ the total density, $\boldsymbol{u} = (u, v, w)^T$ the velocity vector, p the pressure, σ the surface tension coefficient, κ the interface curvature, $\overline{\overline{\tau}}$ the viscous stress tensor,

$$\overline{\overline{\tau}} = \mu \left(\nabla \boldsymbol{u} + (\nabla \boldsymbol{u})^T - \frac{2}{3} (\nabla \cdot \boldsymbol{u}) \overline{\overline{I}} \right) \tag{4.2}$$

with μ the dynamic viscosity and $\overline{\overline{I}}$ the identity tensor, E the specific total energy,

$$E = e + \frac{1}{2} \boldsymbol{u} \cdot \boldsymbol{u} \tag{4.3}$$

where e is the specific internal energy.

4.3.2 Model Closure

The above system of equations is closed by an equation of state, $p = p(\rho, e)$. Here we use the stiffened-gas equation of state (SG-EOS) [57],

$$p = (\gamma - 1)\,\rho e - \gamma p_\infty \tag{4.4}$$

with γ and p_∞ are parameters specific to a given fluid, determined by shockwave Hugoniot data [24, 28]. This equation of state reduces to an ideal gas law when p_∞ is zero and γ then corresponds to the ratio of specific heats. Using the SG-EOS, the mixture speed of sound is given by,

$$c = \sqrt{\gamma \frac{p + p_\infty}{\rho}} \tag{4.5}$$

4.3.3 Mixture Rules

When using a diffuse interface approach, the five-equation model is incomplete in the vicinity of the interface where the fluids are in a mixed state. In order to complete the closure, mixture rules for various quantities need to be defined. Following Allaire, Clerc, and Kokh [3], the mixture rules for volume fraction, density and total internal

energy are given by,

$$\alpha_1 + \alpha_2 = 1 \tag{4.6a}$$

$$\rho_1\alpha_1 + \rho_2\alpha_2 = \rho \tag{4.6b}$$

$$\rho_1\alpha_1 e_1 + \rho_2\alpha_2 e_2 = \rho e \tag{4.6c}$$

The mixture rules for the equation of state parameters are given by,

$$\frac{1}{\gamma - 1} = \alpha_1 \frac{1}{\gamma_1 - 1} + \alpha_2 \frac{1}{\gamma_2 - 1} \tag{4.7a}$$

$$\frac{\gamma p_\infty}{\gamma - 1} = \alpha_1 \frac{\gamma_1 p_{\infty,1}}{\gamma_1 - 1} + \alpha_2 \frac{\gamma_2 p_{\infty,2}}{\gamma_2 - 1} \tag{4.7b}$$

Equation (4.7) ensures that the total energy can be recovered by combining Eqs. (4.3) and (4.4) in both the mixed and single-phase regions,

$$\rho E = \frac{1}{\gamma - 1} p + \frac{\gamma p_\infty}{\gamma - 1} + \frac{1}{2}\rho \boldsymbol{u} \cdot \boldsymbol{u} \tag{4.8}$$

The dynamic viscosity in the mixed region is determined following [123], also used in [28, 45],

$$\mu = \alpha_1\mu_1 + \alpha_2\mu_2 \tag{4.9}$$

where μ_1 and μ_2 are respectively the liquid and gas dynamic viscosities, which are assumed to be constant.

4.4 Numerical Methods

4.4.1 Non-Dimensionalization of Governing Equations

Equations (4.1) to (4.4) describe the evolution of compressible multicomponent flows in dimensional form. Non-dimensionalization of the governing equations is performed by considering three independent reference parameters: ρ_{ref}, x_{ref}, and u_{ref} corresponding to the reference density, reference length scale and reference velocity. The dependent reference parameters are given by,

$$t_{\text{ref}} = \frac{x_{\text{ref}}}{u_{\text{ref}}} \tag{4.10a}$$

$$p_{\text{ref}} = \rho_{\text{ref}}\, u_{\text{ref}}^2 \tag{4.10b}$$

$$E_{\text{ref}} = e_{\text{ref}} = u_{\text{ref}}^2 \tag{4.10c}$$

$$\mu_{\text{ref}} = \rho_{\text{ref}}\, u_{\text{ref}}\, x_{\text{ref}} \tag{4.10d}$$

$$\kappa_{\text{ref}} = \frac{1}{x_{\text{ref}}} \tag{4.10e}$$

$$\sigma_{\text{ref}} = \rho_{\text{ref}}\, u_{\text{ref}}^2\, x_{\text{ref}} \tag{4.10f}$$

The SG-EOS model parameter p_∞ is non-dimensionalized using p_{ref} while γ is already in non-dimensional form. Using the reference parameters described above, the same form as Eqs. (4.1) to (4.4) can be recovered, and will be referred to as the non-dimensional form of the equations henceforth. The corresponding non-dimensional

Reynolds and Weber numbers are given by,

$$\mathrm{Re} = \frac{\rho_{\mathrm{ref}}\, u_{\mathrm{ref}}\, x_{\mathrm{ref}}}{\mu_{\mathrm{ref}}}, \tag{4.11a}$$

$$\mathrm{We} = \frac{\rho_{\mathrm{ref}}\, u_{\mathrm{ref}}^2\, x_{\mathrm{ref}}}{\sigma_{\mathrm{ref}}} \tag{4.11b}$$

4.4.2 Adaptive Mesh Refinement (AMR) Framework

4.4.2.1 Background

In order to meet the mesh resolution requirements of the various aspects of compressible multicomponent flows such as boundary layers, shock wave propagation and the different length scales arising in atomization applications, the governing equations are solved within an adaptive mesh refinement (AMR) framework. In the present work, a novel unstructured Cartesian AMR framework, FARCOM (Fortran Adaptive Refiner for Cartesian Orthogonal Meshes) [7] is used to adaptively discretize the computational domain. The AMR framework produces a fully unstructured mesh with cell-level granularity for isotropic refinement and coarsening operations.

Figure 28 shows an example of the mesh generated by FARCOM in two-dimensional space with four levels of refinement targeted to provide resolution for a given quantity of interest (depicted by solid black line). In general, the framework is capable of generating fully three-dimensional meshes with arbitrary levels of refinement in parallel using the Message Passing Interface (MPI) [44]. The overall refinement and

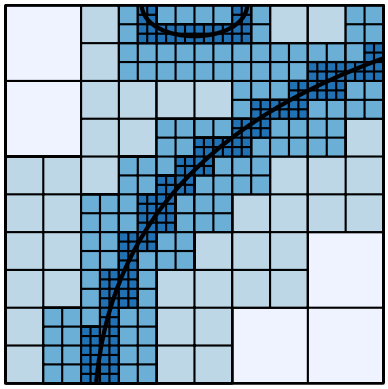
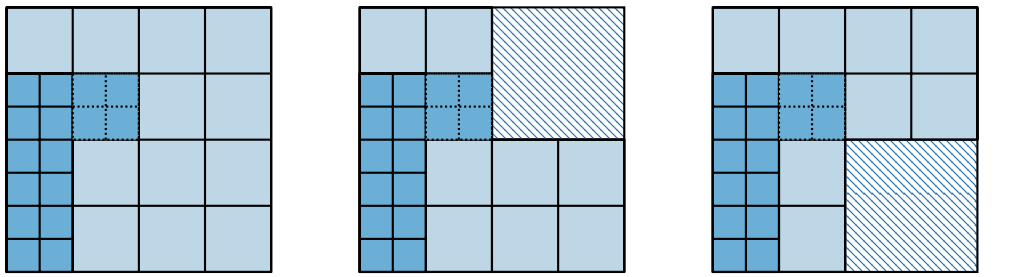


Figure 28: Two-dimensional slice of a quantity of interest (solid black line) resolved by AMR using 4 levels of refinement. The mesh cells are shaded by refinement level.

coarsening criteria is prescribed on a mesh cell level basis, wherein each mesh cell (parent) is further subdivided into 2^{n_d} new cells in n_d-dimensional space (children), which can be observed in Fig. 28. In this regard, FARCOM is similar to quadtree- and octree-based AMR methods [177, 129, 162, 18] as opposed to patch-based or block-based AMR methods [12, 55, 103, 34]. However unlike the octree-based methods, FARCOM does not use tree-type data structures but maintains a fully unstructured data structure.The refinement and coarsening operations are subject to the following two constraints to produce a graded mesh,

1. Cells that share a face (or edge in 2D) with the parent cell may differ by no more than one refinement level.
2. Cells that share an edge (or node in 2D) with the parent cell may differ by no more than one refinement level.

The effect of generating a graded-mesh can be seen in Fig. 28 where individual cells which do not intersect with the quantity of interest are marked for refinement owing to the aforementioned constraints.



(a) Constraints satisfied **(b)** Face-constraint violated **(c)** Edge-constraint violated

Figure 29: Examples of graded AMR meshes with constraints on the refinement operation. Cells are shaded by refinement level; the cell with a pattern is blocking refinement; the cell with dotted lines is the cell to be refined.

The above constraints results in an overall graded mesh and the jump in refinement level across a cell-face (edge in 2D) or cell-edge (node in 2D) is limited to one. Furthermore, it should be noted that the terms "parent" and "children" are for illustration purposes only, as each cell gets removed from the data structure after it undergoes refinement/coarsening and no parent-child relation exists within the data structure. Figure 29 shows an example of a cell undergoing refinement operation (depicted using dotted lines), and the impact of the refinement level of the neighbor cells. In the case of Fig. 29a, all neighbors of the refinement cell have a jump in refinement level of no more than one, and hence is considered a successful refinement

operation. Conversely, in Fig. 29b, one of the face neighbors has a refinement level jump of two (depicted by pattern fill) and is blocking the refinement operation. Similarly, one of the edge neighbors blocks the refinement operation shown in Fig. 29c.

4.4.2.2 Stencils for Higher-Order Schemes

Higher-order spatial schemes used in either the calculation of gradients or for the purposes of interpolation of variables require large stencil sizes. Additionally, these schemes are often developed for one-dimensional, structured grids which may allow for variable spacing in the normal direction but assumes constant spacing in the transverse directions. For the purposes of utilizing higher-order schemes in an unstructured mesh framework, these schemes can be re-derived to suit the present unstructured nature. However, in the present work a general approach to re-creating a one-dimensional, structured grid on the actual unstructured AMR mesh is proposed to re-purpose existing higher-order schemes which have been widely used in literature for shock capturing.

This approach makes use of the operators used to propagate a solution between refinement and coarsening operations as illustrated in Fig. 30. Consider a mesh cell i undergoing refinement to create 2^{n_d} new cells in n_d dimensional space labeled j, $j+1$, $\ldots$, $j+2^{n_d}-1$. The prolongation operator, Θ_P propagates a solution ϕ as,

$$\phi_k = \Theta_P\left(\phi_i\right) = \phi_i, \qquad k = j,\, j+1,\, \ldots,\, j+2^{n_d}-1 \tag{4.12}$$

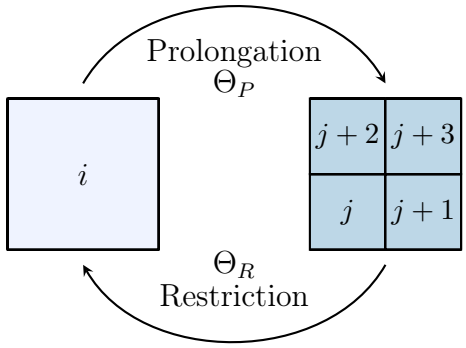


Figure 30: Restriction and prolongation operations used to propagate the solution between coarse and refined cells

which is a conservative operator. For a group of cells $j, j+1, \ldots, j+2^{n_d}-1$ undergoing coarsening to create a new single cell i, the restriction operator, Θ_R propagates a solution ϕ as,

$$\phi_i = \Theta_R(\phi_k) = \frac{1}{2^{n_d}} \sum_k \phi_k, \qquad k = j, j+1, \ldots, j+2^{n_d}-1 \tag{4.13}$$

which results in a conservative volume-averaging of the cells undergoing coarsening.The restriction and prolongation operators can be exploited to map the unstructured mesh data onto a one-dimensional structured stencil. Consider a one-dimensional stencil of size $2n_s$ required to interpolate a quantity ϕ using cell-averaged values from n_s cells on either side of face f,

$$\phi^{n_s}_{\text{stencil},f} = \left(\overline{\phi}_{L-n_s+1}, \overline{\phi}_{L-n_s+2}, \ldots, \overline{\phi}_{L-1}, \overline{\phi}_L, \overline{\phi}_R, \overline{\phi}_{R+1}, \ldots, \overline{\phi}_{R+n_s-2}, \overline{\phi}_{R+n_s-1}\right) \tag{4.14}$$

where $\overline{(\ )}$ indicates a cell-average and L, R indicate cells to the left and right of the given face f. An example stencil with $n_s = 2$ is illustrated in Fig. 31a where the cells shown are used to interpolate a quantity to the cell face f. Depending on the

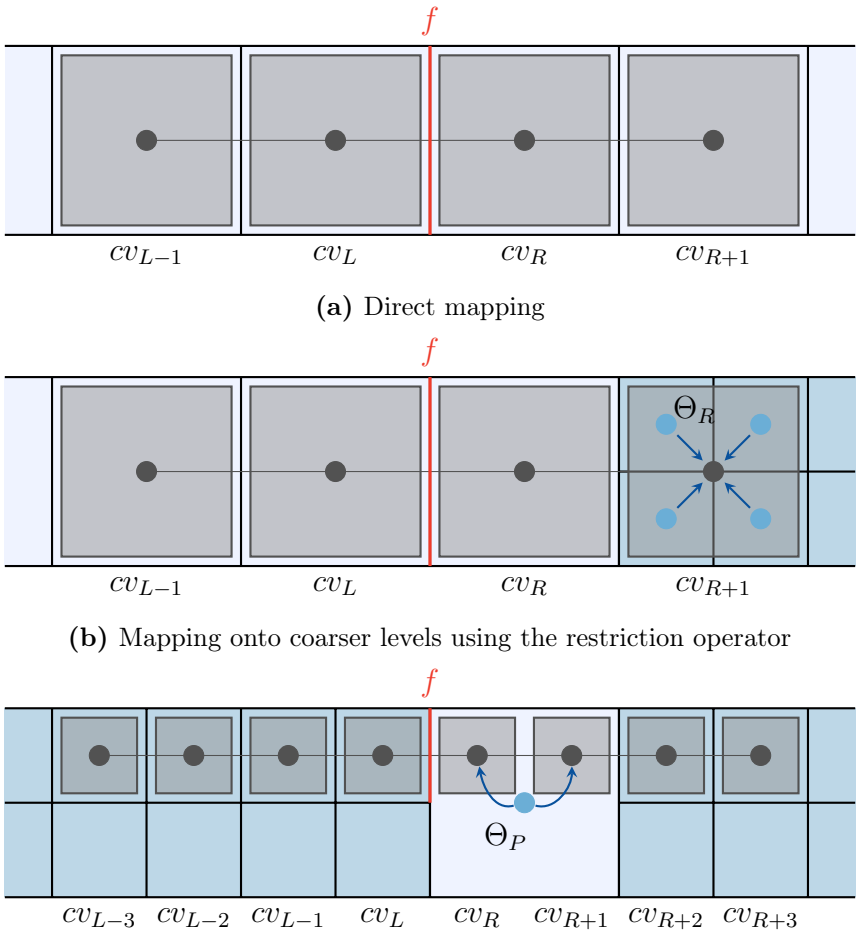
Algorithm 1 Create a one-dimensional stencil for face f

```
1:  procedure StencilCreate(f, φ, n_s)
2:      l_f ← RefinementLevel(f)
3:      for d ← 1, 2 do                          ▷ either normal direction of f
4:          for s ← 1, n_s do                    ▷ n_s is half-stencil size
5:              cv ← NeighborCellIndex(f, d, s)
6:              l_c ← RefinementLevel(cv)
7:              if l_f < l_c then
8:                  φ_stencil[d, s] ← Restrict(φ, cv)        ▷ Eq. (4.13)
9:              else if l_f > l_c then
10:                 φ_stencil[d, s] ← Prolongate(φ, cv)      ▷ Eq. (4.12)
11:             else
12:                 φ_stencil[d, s] ← φ[cv]                  ▷ direct copy
13:             end if
14:         end for
15:     end for
16:     return φ_stencil
17: end procedure
```

refinement level of the cells in the near vicinity of f the numerical approach will vary, as described in the following:

1. If the cells in both normal directions of f are of the same refinement level as f, this indicates that the stencil is already purely one-dimensional and can be used directly. This is illustrated in Fig. 31a where the required gray cells that make up the stencil are obtained directly from the underlying unstructured mesh.
2. If any of the cells in either normal direction of f are at a refinement level of one higher than that of f, the use of the restriction operator Θ_R Eq. (4.13) is mandated. All cells which are at a refinement level higher than f are restricted onto the desired stencil, as shown in Fig. 31b.

3. Conversely, for any cells in the normal direction of f that are at a refinement level of one lower than f, the prolongation operator Θ_P Eq. (4.12) is utilized as shown in Fig. 31c.



(a) Direct mapping

(b) Mapping onto coarser levels using the restriction operator

(c) Mapping onto finer levels using the prolongation operator

Figure 31: Numerical procedure to map unstructured mesh data onto a one-dimensional stencil using the restriction and prolongation operators. Cells shaded in gray indicates the required stencil for the cell face of interest f. Blue shading indicates refinement level.

A general procedure for the creation of stencils is also outlined in Algorithm 1. Using this procedure, stencils of arbitrary sizes can be built using the cell-averaged values of the underlying unstructured mesh and standard numerical schemes developed for structured, Cartesian, equidistant meshes can be utilized. Due to the constraints introduced in Section 4.4.2.1 to produce a graded mesh, the underlying cells in the stencil will never contain a jump in the refinement level of more than one for $n_s \leq 2$ and hence multiple applications of the restriction or prolongation operators are not required. In this work, the band width of each AMR level is chosen that the above observation will hold true for the n_s values required by the schemes described in the following section. Otherwise, the restriction and prolongation operations can be performed recursively to achieve the same effect.

4.4.3 Spatial Discretization

The governing equations Eq. (4.1) can be written in the vectorized form,

$$\frac{\partial \boldsymbol{Q}}{\partial t} + \nabla \cdot (\boldsymbol{F}_{\text{conv}}) = \nabla \cdot (\boldsymbol{F}_{\text{visc}}) + \boldsymbol{S} \tag{4.15}$$

where $\boldsymbol{Q}, \boldsymbol{F}_{\text{conv}}, \boldsymbol{F}_{\text{visc}}, \boldsymbol{S}$ are the vectors of conserved quantities, convective fluxes, viscous fluxes and source terms respectively and are given by,

$$\frac{\partial}{\partial t}\begin{bmatrix}\rho_1\alpha_1\\ \rho_2\alpha_2\\ \rho\boldsymbol{u}\\ \rho E\\ \alpha_1\end{bmatrix} + \nabla\cdot\begin{bmatrix}\rho_1\alpha_1\boldsymbol{u}\\ \rho_2\alpha_2\boldsymbol{u}\\ \rho\boldsymbol{u}\boldsymbol{u}+p\overline{\overline{I}}\\ (\rho E+p)\,\boldsymbol{u}\\ \alpha_1\boldsymbol{u}\end{bmatrix} = \nabla\cdot\begin{bmatrix}0\\ 0\\ \overline{\overline{\tau}}\\ \overline{\overline{\tau}}\boldsymbol{u}\\ 0\end{bmatrix} + \begin{bmatrix}0\\ 0\\ 0\\ \sigma\kappa\nabla\alpha_1\\ \sigma\kappa\nabla\alpha_1\cdot\boldsymbol{u}\\ \alpha_1\,\nabla\cdot\boldsymbol{u}\end{bmatrix} \tag{4.16}$$

The equation for α_1 in Eq. (4.16) is written in a quasi-conservative form of the advection equation Eq. (4.1e), following [74, 28, 45]. Eq. (4.15) is solved using a finite volume discretization, where in each control volume Ω_{cv}, the volume average of the conservative variables $\boldsymbol{Q}$ is stored and is given by,

$$\overline{\boldsymbol{Q}}_{cv} = \frac{1}{\mathcal{V}_{cv}}\int_{\Omega_{cv}} \boldsymbol{Q}\,(\boldsymbol{x})\,d\boldsymbol{x} \tag{4.17}$$

where, $\mathcal{V}_{cv}$ is the volume of control volume Ω_{cv}. The general discretized form of Eq. (4.15) is given by,

$$\frac{d\overline{\boldsymbol{Q}}_{cv}}{dt} = \frac{1}{\Omega_{cv}}\frac{d}{dt}\Big(\sum_f \left(-\boldsymbol{F}_{\text{conv},f} + \boldsymbol{F}_{\text{visc},f}\right)\boldsymbol{n}_f A_f + \overline{\boldsymbol{S}}_{cv}\Big) \tag{4.18}$$

where, the subscript f indicates the faces of the control volume Ω_{cv}, $\boldsymbol{n}_f$ is the outward pointing normal of a given face and A_f is the area of a given face.

4.4.3.1 Convective Terms

The convective fluxes $\boldsymbol{F}_{\text{conv},f}$ are computed following a Godunov approach [52] by defining Riemann problems at each face of a given control volume. Godunov-type schemes are capable of resolving discontinuities without introducing spurious oscillations with a formal first order accuracy; however, they produce large dissipative errors resulting in smearing of discontinuities. This is a consequence of using the control volume averages of $\boldsymbol{Q}$ in the two cells sharing a face as the left and right states of the Riemann problem,

$$\boldsymbol{F}_{\text{conv},f} = \boldsymbol{F}^{\text{Riemann}}\left(\boldsymbol{Q}_f^L, \boldsymbol{Q}_f^R\right) \tag{4.19}$$

Higher-order reconstruction schemes aim to reduce numerical dissipation by minimizing the jump between the left and right states, $\boldsymbol{Q}_f^L$ and $\boldsymbol{Q}_f^R$. As pointed out by several authors [74, 28, 46], performing reconstruction of the conservative variables $\overline{\boldsymbol{Q}}_{cv}$ results in spurious oscillations at the material interface and in regions where the characteristic fields interact strongly. It was shown in [74] that performing reconstructions in the primitive variables form, $\boldsymbol{U} = (\rho_1\alpha_1, \rho_2\alpha_2, \boldsymbol{u}, p, \alpha_1)^T$ avoids these spurious oscillations[7]. In this work, we use a hybrid approach to reconstruct the primitive variables at a given face in order to both resolve discontinuities characteristic of compressible fluid flow (i.e. shocks) and multiphase fluid flow (i.e. material

[7]It was also shown that performing reconstruction on characteristic variables had better results. However, in this work primitive variables are chosen for their relatively lower computational cost.

interfaces),

$$\boldsymbol{U}_f = \begin{cases} \Theta^{\mathrm{HR}}\left(\boldsymbol{U}_{\mathrm{stencil},f}\right), & \text{stencil contains interface cell} \\ \Theta^{\mathrm{PR}}\left(\boldsymbol{U}_{\mathrm{stencil},f}\right), & \text{otherwise} \end{cases} \tag{4.20}$$

where, Θ^{HR} indicates a hyperbolic tangent reconstruction operator and Θ^{PR} indicates a polynomial reconstruction operator for the stencil cells $\boldsymbol{U}_{\mathrm{stencil},f}$ described in Section 4.4.2.2. The hyperbolic tangent reconstruction operator takes precedence in the vicinity of the phase interface and provides a means of limiting the numerical diffusion or smearing of the phase interface, whereas the polynomial reconstruction operator is utilized in regions of flow away from the phase interface to resolve compressible flow dynamics.

The fifth-order accurate weighted essentially non-oscillatory (WENO) [149, 99, 148] reconstruction has been utilized previously [74, 28] in the context of multicomponent compressible flows to create a shock and discontinuity capturing scheme. In the present work, a fifth-order accurate WENO-Z reconstruction of [16], which introduces a lower numerical dissipation relative to the classical WENO scheme is chosen as the polynomial reconstruction operator Θ^{PR}. Additionally, the monotonicity-preserving (MP) bounds of [8] are employed to prevent the reconstruction of unphysical variables. This combination of schemes will be referred to as MP-WENO-Z henceforth. Since the MP-WENO-Z scheme is not total variation diminishing (TVD), the reconstruction might result in values outside the physical range irrespective of mesh spacing sizes. To circumvent this undesired behavior, a Total Variation Diminishing, Monotonic Upstream-centered Scheme for Conservation Laws (TVD-MUSCL) reconstruction [88]

is used instead when the MP-WENO-Z scheme fails. In the worst case scenario when even the TVD-MUSCL reconstruction produces unphysical values, the cell-averaged (CA) values are used for $\boldsymbol{U}_f^L, \boldsymbol{U}_f^R$ resulting in a first-order accurate reconstruction (denoted by the Θ^{CA} operator). This approach for determining the polynomial reconstruction operator (Eq. (4.21)) essentially creates a fallback mechanism which reduces the formal order of accuracy locally in order to deter unphysical values in the reconstruction of $\boldsymbol{U}_f^L, \boldsymbol{U}_f^R$ and as pointed out by [163], the fallback reconstruction does not deteriorate the high-order accuracy in regions of smooth flow features.

$$\Theta^{\text{PR}} = \begin{cases} \Theta^{\text{MP-WENO-Z}}\left(\boldsymbol{U}^3_{\text{stencil},f}\right), & \text{default} \\ \Theta^{\text{TVD-MUSCL}}\left(\boldsymbol{U}^2_{\text{stencil},f}\right), & \text{if } \Theta^{\text{MP-WENO-Z}} \text{ fails} \\ \Theta^{\text{CA}}\left(\boldsymbol{U}^1_{\text{stencil},f}\right), & \text{if } \Theta^{\text{TVD-MUSCL}} \text{ fails} \end{cases} \tag{4.21}$$

The details of the numerical schemes mentioned above are described in C. The condition for a reconstruction scheme to fail is given by,

$$(\rho_1\alpha_1)_f \leq 0 \tag{4.22a}$$

$$(\rho_2\alpha_2)_f \leq 0 \tag{4.22b}$$

$$(\rho_1\alpha_1)_f + (\rho_2\alpha_2)_f \leq 0 \tag{4.22c}$$

$$(\alpha_1)_f < 0 \tag{4.22d}$$

$$(\alpha_1)_f > 1 \tag{4.22e}$$

where $(\ )_f$ refers to a reconstructed value at a cell face using any reconstruction scheme. It should be noted that there is no upper-bound on the phasic densities, unlike for the liquid volume fraction.

Although the polynomial reconstruction operator Eq. (4.21) provides a good approach for robust shock-capturing, it does not prevent the excessive smearing of phase interfaces. For this reason, the Tangent of Hyperbola for INterface Capturing (THINC) scheme [155, 175, 45] is chosen as the hyperbolic tangent operator Θ^{HR} to reconstruct the face values in regions near the phase interface,

$$\Theta^{\mathrm{HR}} = \Theta^{\mathrm{THINC}}\left(\phi^2_{\mathrm{stencil},f}\right) \tag{4.23}$$

The details of the THINC operator are outlined in C. The hyperbolic tangent reconstruction is applied on cells which contain a phase interface determined by [155, 45],

$$\left(\overline{\alpha_1}\right)_{cv} \in \left(\epsilon_{\mathrm{THINC}}, 1 - \epsilon_{\mathrm{THINC}}\right) \tag{4.24}$$

also subject to a monotonicity constraint given by [155, 45],

$$\left(\left(\overline{\alpha_1}\right)_R - \left(\overline{\alpha_1}\right)_L\right) \cdot \left(\left(\overline{\alpha_1}\right)_L - \left(\overline{\alpha_1}\right)_{L-1}\right) > \epsilon_{\mathrm{MC}} \tag{4.25}$$

with the values of $\epsilon_{\mathrm{THINC}} = 1 \times 10^{-4}$ and $\epsilon_{\mathrm{MC}} = 1 \times 10^{-12}$. If the THINC reconstruction results in unphysical quantities as determined by Eq. (4.22), then the reconstruction using the polynomial reconstruction operator Eq. (4.21) is used instead. The overall approach to reconstructing the left and right states $\boldsymbol{U}_f^L, \boldsymbol{U}_f^R$ for a given cell face f is outlined in Algorithm 2. Using the approach described above to create the left and right states $\boldsymbol{U}_f^L, \boldsymbol{U}_f^R$ for a given face f, a Riemann problem Eq. (4.19) is solved using an approximate Riemann solver following [46, 45], the details of which are described in D.

Algorithm 2 Reconstruct $\boldsymbol{U}$ at cell face f

procedure FACERECONSTRUCTION($f, \boldsymbol{U}$)
  $\boldsymbol{U}^3_{\text{stencil}} \leftarrow$ STENCILCREATE $(f, \boldsymbol{U}, 3)$ ▷ Algorithm 1
  $\boldsymbol{U}^2_{\text{stencil}} \leftarrow$ STENCILCREATE $(f, \boldsymbol{U}, 2)$
  $\boldsymbol{U}^1_{\text{stencil}} \leftarrow$ STENCILCREATE $(f, \boldsymbol{U}, 1)$
  $\boldsymbol{U}_f \leftarrow$ MP-WENO-Z $(\boldsymbol{U}^3_{\text{stencil}})$ ▷ Eq. (4.21)
  if RECONSTRUCTIONFAIL $(\boldsymbol{U}_f)$ **then** ▷ Eq. (4.22)
    $\boldsymbol{U}_f \leftarrow$ TVD-MUSCL $(\boldsymbol{U}^2_{\text{stencil}})$ ▷ Eq. (4.21)
    if RECONSTRUCTIONFAIL $(\boldsymbol{U}_f)$ **then** ▷ Eq. (4.22)
      $\boldsymbol{U}_f \leftarrow$ CA $(\boldsymbol{U}^1_{\text{stencil}})$ ▷ Eq. (4.21)
    end if
  end if
  $\boldsymbol{U}_f \leftarrow$ THINC $(\boldsymbol{U}^2_{\text{stencil}})$ ▷ Eq. (4.23)
  return $\boldsymbol{U}_f$
end procedure

4.4.3.2 Diffusive Terms

The vector of viscous fluxes $\boldsymbol{F}_{\text{visc}}$ in Eq. (4.15) requires gradients of the velocity vector $\boldsymbol{u}$. As described initially by [123, 74] and followed later by [28, 46], the velocity used in the viscous terms should be identical to the one used in the convective flux for consistency reasons. Applying the definition of the cell-average Eq. (4.17) to the viscous flux vector the discretized form can be written as,

$$\nabla \cdot (\boldsymbol{F}_{\text{visc}}) = \frac{1}{\Omega_{cv}} \sum_f \begin{pmatrix} 0 \\ 0 \\ \overline{\overline{\tau}}_f \\ \overline{\overline{\tau}}_f \, \boldsymbol{u}_f \\ 0 \end{pmatrix} \cdot \boldsymbol{n}_f A_f \tag{4.26}$$

where the required velocity vector at the face $\boldsymbol{u}_f$ is calculated using Eq. (D.6). The viscous stress tensor at the face $\overline{\overline{\tau}}_f$ requires the computation of the gradient of velocity at the face $\nabla \boldsymbol{u}_f$, see Eq. (4.2). Following [123, 28, 45], it is calculated using,

$$\nabla \boldsymbol{u}_f = \frac{1}{2}\Big((\nabla \boldsymbol{u})_L + (\nabla \boldsymbol{u})_R \Big) \tag{4.27}$$

where $(\nabla \boldsymbol{u})_L, (\nabla \boldsymbol{u})_R$ denote the gradients at the cells sharing the cell face f. These gradients at the cells are calculated using the upwinded velocities Eq. (D.6),

$$(\nabla u)_{cv} = \frac{1}{\Omega_{cv}} \sum_f u_f \boldsymbol{n}_f A_f \tag{4.28a}$$

$$(\nabla v)_{cv} = \frac{1}{\Omega_{cv}} \sum_f v_f \boldsymbol{n}_f A_f \tag{4.28b}$$

$$(\nabla w)_{cv} = \frac{1}{\Omega_{cv}} \sum_f w_f \boldsymbol{n}_f A_f \tag{4.28c}$$

4.4.3.3 Source Terms

The vector of source terms $\boldsymbol{S}$ in Eq. (4.15) requires the discretization of the terms $\sigma\kappa\nabla\alpha_1$, $\sigma\kappa\nabla\alpha_1 \cdot \boldsymbol{u}$, $\alpha_1 \nabla \cdot \boldsymbol{u}$. Similar to the viscous terms, the velocity used should be identical to the velocity in the convective flux. To aid with this, the $\sigma\kappa\nabla\alpha_1 \cdot \boldsymbol{u}$ term is re-cast into $\sigma\kappa\big(\nabla \cdot (\alpha_1 \boldsymbol{u}) - \alpha_1 \nabla \cdot \boldsymbol{u}\big)$ using chain-rule to retrieve the convective form. Applying the definition of the cell-average Eq. (4.17) to the source term vector,

$$\overline{\boldsymbol{S}}_{cv} = \frac{1}{\mathcal{V}_{cv}} \int_{\Omega_{cv}} \boldsymbol{S}(\boldsymbol{x})\, d\boldsymbol{x}, \tag{4.29}$$

the discretized form can be written as,

$$\overline{\boldsymbol{S}}_{cv} = \frac{1}{\Omega_{cv}} \sum_{f} \begin{pmatrix} 0 \\ 0 \\ \overline{\sigma\kappa}\,\alpha_{1,f} \\ \overline{\sigma\kappa}\Big((\alpha_1 \boldsymbol{u})_f - \overline{\alpha_1}\boldsymbol{u}_f \Big) \\ \overline{\alpha_1}\,\boldsymbol{u}_f \end{pmatrix} \cdot \boldsymbol{n}_f A_f \tag{4.30}$$

where the required volume fraction at the face $\alpha_{1,f}$ is calculated using Eq. (D.7) and $\boldsymbol{u}_f$ is calculated using Eq. (D.6) as done with the diffusive terms.

4.4.4 Interfacial Geometric Properties

The phase interface within a volume fraction field is often uniquely identified using calculated geometric properties such as the interface normal vector $\boldsymbol{n} = (n^x, n^y, n^z)^T$ and the interface curvature κ. These quantities are also required in the computation of surface tension terms and in the THINC reconstruction as described in previous sections. The interface normal vector is given by,

$$\boldsymbol{n} = -\frac{\nabla \alpha_1}{|\nabla \alpha_1|} \tag{4.31}$$

However, computing gradients on the smooth but almost discontinuous volume fraction field α_1 due to the application of THINC reconstruction is ill-advised. To circumvent this, a mollified volume fraction field ψ is used following the approach of Shukla, Pantano, and Freund [150] and later by [46, 45] to estimate the gradients of

α_1,

$$\psi = \frac{\alpha_1^{\omega}}{\alpha_1^{\omega} + (1-\alpha_1)^{\omega}}, \qquad \text{for } \omega < 1. \tag{4.32}$$

with a constant value of $\omega = 0.1$ [150]. The gradient of the volume fraction is then estimated using,

$$\nabla\alpha_1 = \frac{\nabla\psi}{\omega}\big(\alpha_1\,(1-\alpha_1)\,\big)^{1-\omega}\,\big(\alpha_1^{\omega} + (1-\alpha_1)^{\omega}\,\big)^2 \tag{4.33}$$

The estimation of interface curvature is described in the following. It should be noted that within the present AMR framework, the interface is always maintained at the finest refinement level and throughout this work the width of the finest level is guaranteed to span the desired region of curvature estimation. The consequence of creating this constraint is that there are no refinement level jumps within the context of curvature estimation and therefore, the remainder of this section is described in the context of a structured, equidistant, Cartesian mesh.

The height function (HF) method [29] for estimating the interface curvature, was originally proposed for methods where the interface is resolved within one mesh cell and does not spread over a number of cells. The position of the interface is recovered using discrete sums of the volume fraction along the direction of the maximum component of the normal vector. In three-dimensional Cartesian space it is given by,

$$H_{i,j,k} = \begin{cases} \sum_{i'=i-n_{\mathrm{HF}}}^{i+n_{\mathrm{HF}}} \alpha_{i',j,k}\,\Delta x & \text{for } \left|n^x_{i,j,k}\right| > \left|n^y_{i,j,k}\right| \text{ and } \left|n^x_{i,j,k}\right| > \left|n^z_{i,j,k}\right| \\ \sum_{j'=j-n_{\mathrm{HF}}}^{j+n_{\mathrm{HF}}} \alpha_{i,j',k}\,\Delta y & \text{for } \left|n^y_{i,j,k}\right| > \left|n^x_{i,j,k}\right| \text{ and } \left|n^y_{i,j,k}\right| > \left|n^z_{i,j,k}\right| \\ \sum_{k'=k-n_{\mathrm{HF}}}^{k+n_{\mathrm{HF}}} \alpha_{i,j,k'}\,\Delta z & \text{for } \left|n^z_{i,j,k}\right| > \left|n^x_{i,j,k}\right| \text{ and } \left|n^z_{i,j,k}\right| > \left|n^y_{i,j,k}\right| \end{cases} \tag{4.34}$$

for HF columns of total height $2n_{\mathrm{HF}} + 1$. Traditionally for use with sharp-interface methods, n_{HF} is chosen as 3, see Fig. 32a. In the present work n_{HF} is chosen as 5 to compensate for the increased thickness of the interface, typical of diffuse-interface methods (Fig. 32b).

Once the interface height is approximated in the height columns using Eq. (4.34), the interface curvature can be calculated using,

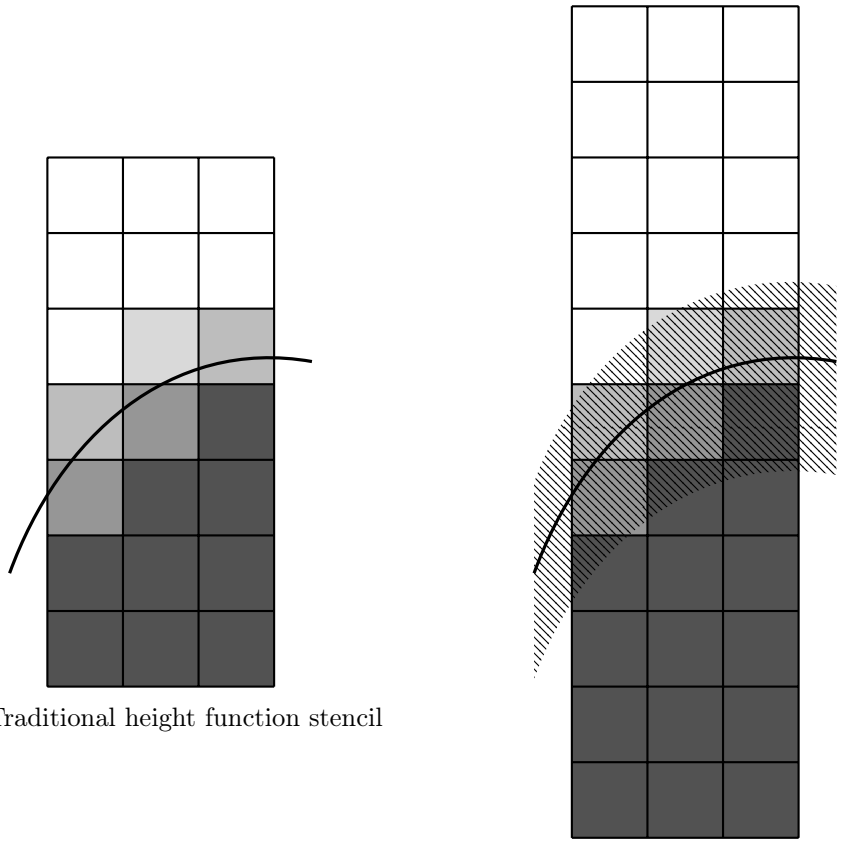
$$\kappa = \begin{cases} \kappa_x, & \text{if } \left|n^x_{i,j,k}\right| > \left|n^y_{i,j,k}\right| \text{ and } \left|n^x_{i,j,k}\right| > \left|n^z_{i,j,k}\right| \\ \kappa_y, & \text{if } \left|n^y_{i,j,k}\right| > \left|n^x_{i,j,k}\right| \text{ and } \left|n^y_{i,j,k}\right| > \left|n^z_{i,j,k}\right| \\ \kappa_z, & \text{if } \left|n^z_{i,j,k}\right| > \left|n^x_{i,j,k}\right| \text{ and } \left|n^z_{i,j,k}\right| > \left|n^y_{i,j,k}\right| \end{cases} \tag{4.35}$$

with,

$$\kappa_x = \frac{\partial_{yy}H + \partial_{zz}H + \partial_{yy}H\,(\partial_z H)^2 + \partial_{zz}H\,(\partial_y H)^2 - \partial_{yz}H\,\partial_y H\,\partial_z H}{\left(1 + (\partial_y H)^2 + (\partial_z H)^2\right)^{3/2}} \tag{4.36a}$$

$$\kappa_y = \frac{\partial_{xx}H + \partial_{zz}H + \partial_{xx}H\,(\partial_z H)^2 + \partial_{zz}H\,(\partial_x H)^2 - \partial_{xz}H\,\partial_x H\,\partial_z H}{\left(1 + (\partial_x H)^2 + (\partial_z H)^2\right)^{3/2}} \tag{4.36b}$$

$$\kappa_z = \frac{\partial_{xx}H + \partial_{yy}H + \partial_{xx}H\,(\partial_y H)^2 + \partial_{yy}H\,(\partial_x H)^2 - \partial_{xy}H\,\partial_x H\,\partial_y H}{\left(1 + (\partial_x H)^2 + (\partial_y H)^2\right)^{3/2}} \tag{4.36c}$$



(a) Traditional height function stencil

(b) Extended height function stencil

Figure 32: Height function stencil used for estimating interface curvature in two-dimensional space. Cells are shaded by volume fraction. The thick line represents the phase interface for a sharp-interface method and the patterned area represents the diffused interface region.

where $\partial_k, \partial_{kk}$ represent the first- and second-derivatives of the HF columns in the $k-$direction, which are computed using central finite-difference operators. In three-dimensional space however, the smoothed partial derivatives of López et al. [102] are used instead to improve accuracy, as shown in the results in the following sections.

By construction principle, the HF technique described above only provides an estimate of the curvature at mesh cells which contain the height-function columns [29]. Therefore, a redistribution scheme is used following the above procedure to determine the curvature values at the neighboring cells of the diffused region. Within the diffuse region, the interface curvature values are constant in the interface normal direction and can therefore be described by the hyperbolic boundary-value problem (BVP),

$$\boldsymbol{a}(\boldsymbol{x}) \cdot \nabla \kappa(\boldsymbol{x}) = 0 \tag{4.37}$$

using the curvature estimate from the height function method as boundary conditions. The characteristic velocity of the hyperbolic equation Eq. (4.37) is,

$$\boldsymbol{a} = \begin{cases} \boldsymbol{n}, & \text{if } \alpha_1 \leq 0.5, \\ -\boldsymbol{n}, & \text{if } \alpha_1 > 0.5 \end{cases} \tag{4.38}$$

Equation (4.37) is traditionally solved using either the fast marching method (FMM) [145] or the fast sweeping method (FSM) [5]; in this work, the latter is used. A first-order upwind discretization of the interface curvature with respect to the characteristic velocity in the $x-$direction is given by,

$$\partial_x \kappa_{i,j,k} = \frac{1 + \text{sign}\left(a^x_{i,j,k}\right)}{2} \left(\frac{\Delta^-_x \kappa_{i,j,k}}{\Delta x} \right) + \frac{1 - \text{sign}\left(a^x_{i,j,k}\right)}{2} \left(\frac{\Delta^+_x \kappa_{i,j,k}}{\Delta x} \right) \tag{4.39}$$

where Δ^-_x, Δ^+_x indicates the forward and backward undivided finite differences. The upwind discretization for $y-$ and $z-$directions follows in analogous fashion. Using the upwind discretization in Eq. (4.39), a matrix-free iterative Gauss-Seidel scheme

can be recovered for Eq. (4.37),

$$\kappa_{i,j,k}^{(n+1)} = \frac{1}{2} \frac{\frac{1}{\Delta x}\left\langle a_{i,j,k}^{x} \kappa_{i,j,k}^{(n)} \right\rangle + \frac{1}{\Delta y}\left\langle a_{i,j,k}^{y} \kappa_{i,j,k}^{(n)} \right\rangle + \frac{1}{\Delta z}\left\langle a_{i,j,k}^{z} \kappa_{i,j,k}^{(n)} \right\rangle}{\frac{1}{\Delta x}\left|a_{i,j,k}^{x}\right| + \frac{1}{\Delta y}\left|a_{i,j,k}^{y}\right| + \frac{1}{\Delta z}\left|a_{i,j,k}^{z}\right|} \tag{4.40}$$

with

$$\left\langle a_{i,j,k}^{x} \kappa_{i,j,k}^{(n)} \right\rangle = a_{i,j,k}^{x}\left(\kappa_{i-1,j,k}^{(n)} - \kappa_{i+1,j,k}^{(n)}\right) + \left|a_{i,j,k}^{x}\right|\left(\kappa_{i-1,j,k}^{(n)} + \kappa_{i+1,j,k}^{(n)}\right) \tag{4.41}$$

and likewise for the $y-$ and $z-$directions, where (n) denotes the iteration number. The iterative scheme is performed until the following stopping criterion [5] is reached,

$$\left|\kappa^{(n)} - \kappa^{(n-1)}\right|_{\infty} \leq \epsilon_{\text{FSM}} \tag{4.42}$$

with the error tolerance set to $\epsilon_{\text{FSM}} = 1 \times 10^{-12}$ throughout this work. The iterative FSM is applied only on a narrow band in the neighborhood of the diffuse interface given by $\alpha_1 \in (\epsilon_{\text{THINC}}, 1 - \epsilon_{\text{THINC}})$.

4.4.5 Temporal Discretization

With the spatial discretization described in Section 4.4.3 applied, Eq. (4.18) can be re-cast to form a system of ordinary differential equations (ODEs),

$$\frac{\mathrm{d}\boldsymbol{Q}}{\mathrm{d}t} = \mathcal{L}\left(\boldsymbol{Q}\right) \tag{4.43}$$

where $\mathcal{L}$ denotes the discrete spatial differential operator which consists of the convective, viscous and source terms of Eq. (4.18). The system of ODEs in Eq. (4.43) is discretized in time using a low-storage variant [80] of the three stage, third-order

strong stability-preserving Runge-Kutta scheme of Shu and Osher [149],

$$\boldsymbol{Q}^{(1)} = \boldsymbol{Q}^{n} + \Delta t\, \mathcal{L}\left(\boldsymbol{Q}^{n}\right) \tag{4.44a}$$

$$\boldsymbol{Q}^{(2)} = \frac{3}{4}\boldsymbol{Q}^{n} + \frac{1}{4}\boldsymbol{Q}^{(1)} + \frac{1}{4}\Delta t\, \mathcal{L}\left(\boldsymbol{Q}^{(1)}\right) \tag{4.44b}$$

$$\boldsymbol{Q}^{n+1} = \frac{1}{3}\boldsymbol{Q}^{n} + \frac{2}{3}\boldsymbol{Q}^{(2)} + \frac{2}{3}\Delta t\, \mathcal{L}\left(\boldsymbol{Q}^{(2)}\right) \tag{4.44c}$$

to time advance the solution from time-level n to a time-level $n+1$ using a time-step size Δt. The time-step size is limited using the CFL condition,

$$\Delta t = \mathrm{CFL} \times \min\left(\Delta t_{\mathrm{conv}},\, \Delta t_{\mathrm{visc}},\, \Delta t_{\mathrm{cap}}\right) \tag{4.45}$$

where Δt_{conv} is the convective time-step criterion [165],

$$\Delta t_{\mathrm{conv}} = \min_{cv}\left(\frac{\Delta_{cv,\mathrm{min}}}{\max\left(|u|, |v|, |w|\right) + c}\right) \tag{4.46}$$

and Δt_{visc} is the viscous time-step criterion,

$$\Delta t_{\mathrm{visc}} = \min_{cv}\left(\frac{\Delta^{2}_{cv,\mathrm{min}}}{\mu}\right) \tag{4.47}$$

and Δt_{cap} is the capillary time-step criterion [17] due to the presence of surface tension forces,

$$\Delta t_{\mathrm{cap}} = \min_{cv}\left(\frac{\left(\rho_1 + \rho_2\right)\Delta^{3}_{cv,\mathrm{min}}}{4\pi\sigma}\right) \tag{4.48}$$

with $\Delta_{cv,\mathrm{min}}$ denoting the minimum cell size.

4.5 Results and Discussion

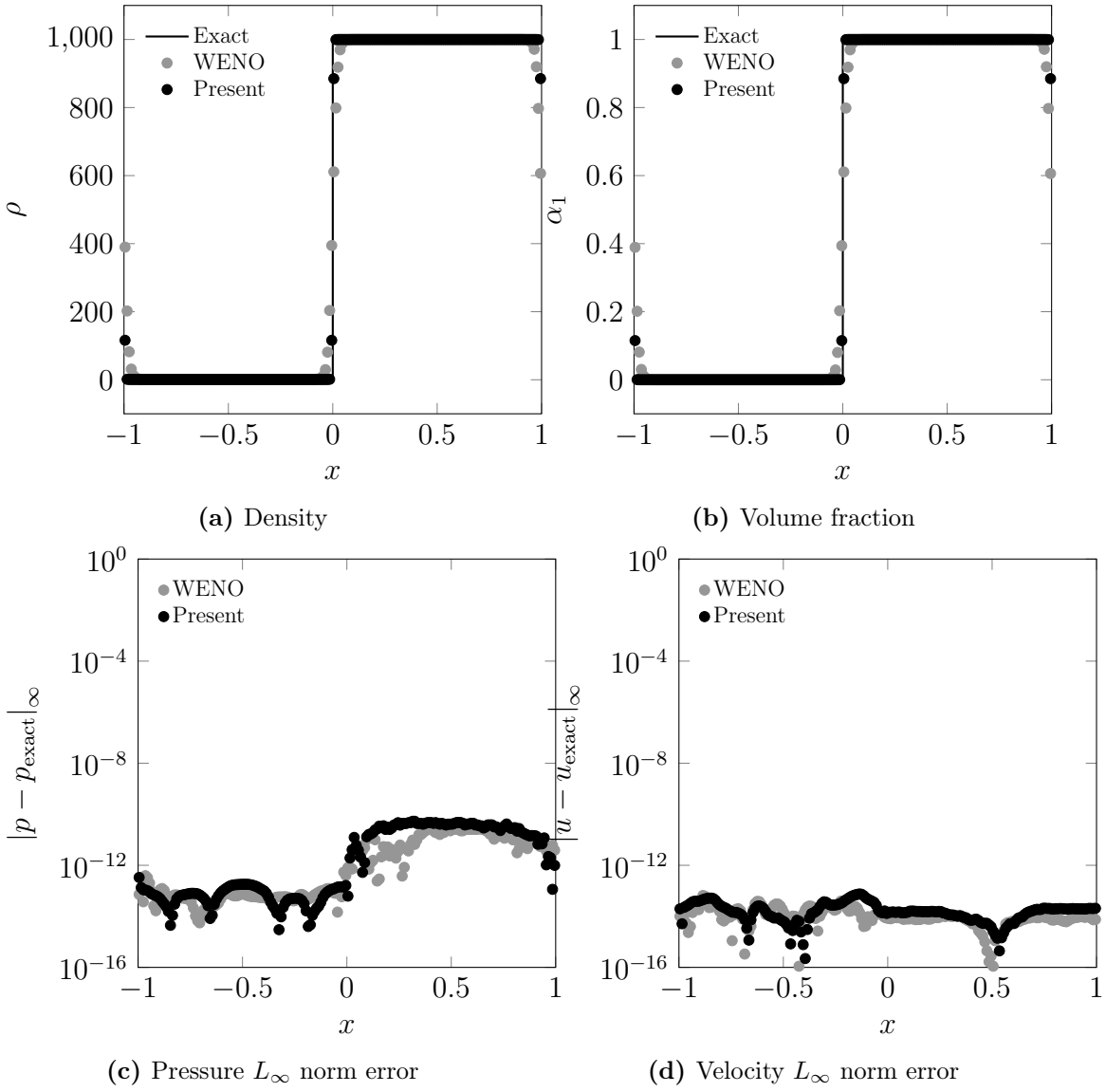
4.5.1 One-Dimensional Advection of an Isolated Material Front

The pure advection of an isolated material front in one-dimension is a problem proposed initially by [2] and used heavily in relevant literature [74, 28, 45, 78] to illustrate the observance and severity of spurious oscillations at the phase interface. A gas-liquid interface is initialized in a $[-1, 1]$ one-dimensional domain with the following non-dimensional conditions,

$$(\rho_1\alpha_1, \rho_2\alpha_2, u, p, \alpha_1)^T = \begin{cases} (\quad 0, 1, 0.5, 1, 0)^T, & \text{if } -1 \leq x < 0 \\ (1000, 0, 0.5, 1, 1)^T, & \text{if } \quad 0 \leq x \leq 1 \end{cases} \tag{4.49}$$

with the non-dimensional SG-EOS material properties for the liquid given by $(\gamma_1, p_{\infty,1})^T = (4.4, 6000)^T$, and for the gas by $(\gamma_2, p_{\infty,2})^T = (1.4, 0)^T$ following [45]. No surface tension or viscous effects are considered for this problem. The domain is discretized using an equidistant mesh spacing of 0.01, while periodic boundary conditions are applied on either end of the domain. The initial condition is advanced in time to $t = 4$ using a time-step value dictated by CFL $= 0.6$, which results in one-period of advection in the periodic domain.

Figure 33 shows the numerical solution computed using only a WENO reconstruction and also by the approach described in Section 4.4.3 and C. In both cases, spurious oscillations are not observed at the material interface and the errors in



(a) Density

(b) Volume fraction

(c) Pressure L_∞ norm error

(d) Velocity L_∞ norm error

Figure 33: Numerical solution for the one-dimensional advection problem at $t = 4$

pressure and velocity are of the same order, roughly in the vicinity of numerical round-off. The present approach to reconstruct the face values however, is able to

preserve the interface thickness unlike the pure WENO reconstruction, indicating that the THINC reconstruction is vital to reducing numerical diffusion even when used in conjunction with a high-order scheme.

4.5.2 Verification of Calculated Curvature

The interface curvature calculated using the modified height function method in Section 4.4.4 is verified by comparing to the exact curvature for a two-dimensional circle and a three-dimensional sphere. The computational domain is $[-1, 1]$ in all directions considered, with the center of the unit circle/sphere located at the origin. The infinity norm errors are computed for the curvature calculated using the standard partial derivatives and the smoothed partial derivatives of López et al. [102]. It should be noted that the smoothed partial derivatives are only valid for the three-dimensional case.

Table 11: Infinity norm errors and convergence order in the estimation of curvature for a static circle and sphere

Cells per diameter	Standard (2D)		Standard (3D)		Smoothed (3D)	
	L_∞ norm	Order	L_∞ norm	Order	L_∞ norm	Order
16	5.97×10^{-2}	-	4.45×10^{-1}	-	1.18×10^{-1}	-
32	1.36×10^{-2}	2.13	5.02×10^{-2}	3.15	2.03×10^{-2}	2.55
64	2.97×10^{-3}	2.20	1.57×10^{-2}	1.68	5.04×10^{-3}	2.01
128	7.60×10^{-4}	1.97	2.80×10^{-3}	2.48	1.32×10^{-3}	1.94
256	1.86×10^{-4}	2.03	7.84×10^{-4}	1.84	3.26×10^{-4}	2.01

The infinity norm errors of calculated curvature shown in Table 11 indicate a second-order convergence of the method used. Finally, the magnitude of errors when using the smoothed partial derivatives of [102] are relatively smaller.

4.5.3 Surface Tension Verification Tests

4.5.3.1 Static Inviscid Drop with Exact Prescribed Curvature

To verify the implementation of the surface tension terms, it is desirable to check initially if the model is able to perform using an exact prescribed curvature instead of using the method described in Section 4.4.4. To this end, the canonical test case suggested by Williams, Kothe, and Puckett [172] of a static inviscid drop in static equilibrium is performed to verify that the surface tension forces discretely balance the pressure jump across the interface, resulting in the absence of spurious oscillations. A drop (column in 2D, sphere in 3D) of radius 0.5 is placed at the center of a 2^2 (or 2^3) domain with a constant static mesh spacing of $(D/\Delta) = 16$. The surface tension coefficient is set to unity and the pressure is set such that the pressure jump $\Delta p = p_1 - p_2 = \sigma\kappa$. The SG-EOS parameters for both the fluids are set to $(\gamma, p_\infty)^T = (1.4, 0)^T$ for varying density ratios (with unit ρ_1). The solution is advanced in time for a single time-step with a time-step value of $\Delta t = 10^{-6}$.

Table 12: Errors in velocity and kinetic energy after single time-step for a static equilibrium drop using exact prescribed curvature

$\frac{\rho_1}{\rho_2}$	Column		Sphere	
	$L_\infty(\|\boldsymbol{u}\|)$	E_{kin}	$L_\infty(\|\boldsymbol{u}\|)$	E_{kin}
10^{-8}	4.55×10^{-20}	1.05×10^{-42}	1.04×10^{-19}	1.05×10^{-40}
10^{-4}	4.77×10^{-20}	1.48×10^{-42}	1.03×10^{-19}	1.81×10^{-40}
10^{-2}	5.02×10^{-20}	1.80×10^{-42}	1.04×10^{-19}	2.20×10^{-40}
1	4.49×10^{-20}	2.33×10^{-42}	7.25×10^{-20}	2.57×10^{-40}
10^{2}	1.83×10^{-18}	7.77×10^{-41}	2.27×10^{-18}	6.78×10^{-39}
10^{4}	1.99×10^{-16}	5.93×10^{-39}	2.35×10^{-16}	6.28×10^{-37}
10^{8}	1.66×10^{-12}	4.77×10^{-35}	2.42×10^{-12}	4.73×10^{-33}

Table 12 shows the computed errors in the velocity and the kinetic energy for both the column and sphere case after one time step. It can be seen that the errors are of the order of machine precision even for high density ratios. It should be noted that the error in the maximum pressure jump in the domain was observed to be 1.78×10^{-15} for the column case and 3.73×10^{-14} for the sphere case, for all density ratios. Thus the discretization of the surface tension terms discretely balances the surface tension and pressure jump across the interface, provided the exact curvature is known.

4.5.3.2 Inviscid Oscillation of a Column

To further verify the implementation of surface tension, it is also desirable to study its performance under dynamic conditions, such as the inviscid oscillation of a two-dimensional column. A column of radius $r = 0.5$ is placed at the center of a

2^2 domain with both the fluid within the column and surrounding fluid initially at rest. The column is initialized with a mode $n = 2$ perturbation with initial amplitude of $0.01r$. The theoretical oscillation period within the linear regime for an ellipse in two-dimensional space is given by [85],

$$T = 2\pi \sqrt{\frac{(\rho_1 + \rho_2)\, r^3}{n\,(n^2 - 1)\,\sigma}} \tag{4.50}$$

with the surface tension coefficient $\sigma = 1$. The fluid within the column is initialized with $(\rho_1, p_1, \gamma_1, p_{\infty,1})^T = (1, p_2 + \sigma\kappa, 4.4, 6000)^T$ and the surrounding fluid with $(\rho_2, p_2, \gamma_2, p_{\infty,2})^T = (0.01, 1, 1.4, 0)^T$. Static meshes with constant mesh spacings of $(2r/\Delta) = 64, 128$ are used to perform the simulation.

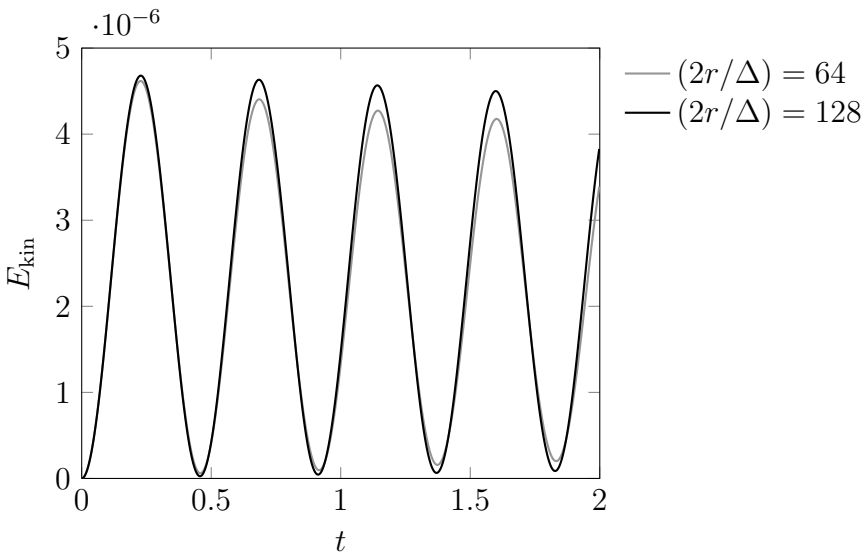


Figure 34: Integral kinetic energy over time for the zero gravity oscillation of a column test case

The temporal evolution of the integral kinetic energy is given in Fig. 34 for the two mesh spacings. It can be seen that the kinetic energy gradually decays over time due to the presence of numerical dissipation, since no physical dissipation is taken into account for the inviscid case. The exact analytical oscillation period for this case is $T_{\text{exact}} = 0.911$, as given by Eq. (4.50). The calculated oscillation period for the two mesh sizes presented are $T_{64} = 0.916$ and $T_{128} = 0.913$, resulting in a order of convergence of 1.26. Therefore, the present implementation shows between a first and second order convergence of surface tension under mesh refinement, which is comparable to those presented in [61, 78, 166].

4.5.4 Shock-Interface Interaction Tests

In this section, several problems featuring the interaction of shocks with interfaces are studied with variations in shock Mach numbers and fluids considered. These tests are good candidates to test the performance of the AMR framework to capture various quantities of interest. For this purpose, all subsequent tests undergo mesh refinement changes based on the following criteria:

1. The fluid which is impacted by the incident shock is maintained at the highest level of refinement. This is performed by marking mesh cells based on magnitude of the volume fraction using $\alpha_1 > \epsilon_{\text{THINC}}$.

2. Shocks in the domain are maintained at the highest level of refinement. This is performed by marking mesh cells based on the gradient of pressure using $\nabla p > p^{\text{AMR}}$.

The refinement levels of all other mesh cells are propagated from the cell marked using the above criteria by means of a band marching algorithm [7]. This results in high-level of refinement in regions of interest and aggressive coarsening of mesh cells away from these regions.

Visualization of the numerical solution can be performed using numerical schlieren to highlight gradients of density. In this work, the idealized schlieren images proposed by Quirk and Karni [133] are utilized,

$$\phi_{\text{sch}} = \exp\left(-k\frac{|\nabla\rho|}{|\nabla\rho|_{\max}}\right) \tag{4.51}$$

where $k = k_1\alpha_1 + k_2\alpha_2$ is a scaling parameter, which allows for the visualization of wave patterns in multicomponent flows with vastly different material parameters by adjusting k_1, k_2 appropriately. Finally, all the boundaries in the computational domain employ a zeroth-order non-reflecting extrapolation boundary conditions [93].

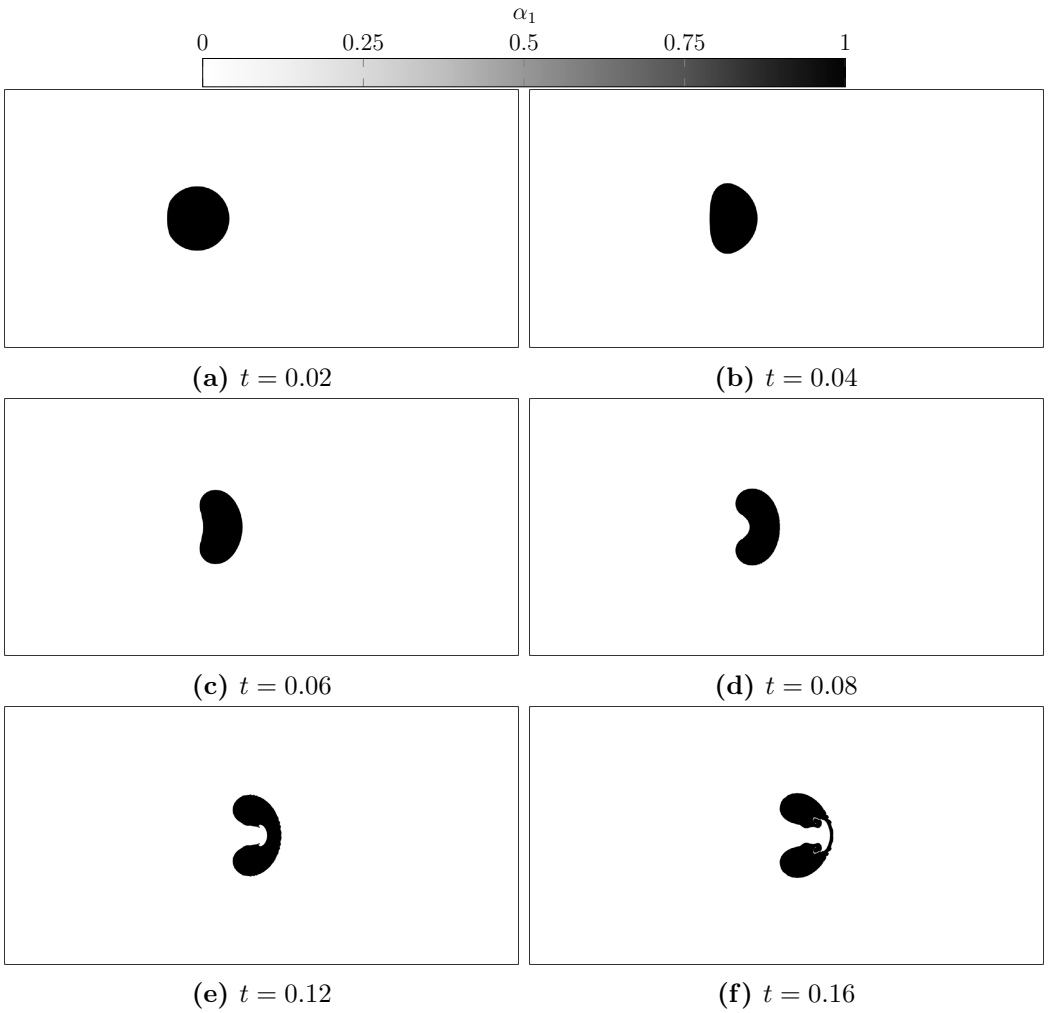
4.5.4.1 Ma = 1.22 Shock in Air Impacting a Helium Column

The case of a shock in air interacting with a helium column initially at rest following the experiments of Haas and Sturtevant [56] is studied. A column of diameter $D = 0.05\,\text{m}$ with its center at $(0.0075\,\text{m}, 0\,\text{m})$ is placed surrounded by air in

a rectangular two-dimensional domain of size $[0\,\mathrm{m}, 0.02\,\mathrm{m}] \times [-0.005\,\mathrm{m}, 0.005\,\mathrm{m}]$. A planar shock of Mach number $\mathrm{Ma} = 1.22$ under atmospheric pressure ($p = p_1 = p_2 = 101\,325\,\mathrm{Pa}$) is placed at $x = 0.005\,\mathrm{m}$ moving left to right. The initial condition in the pre-shock air is set to $(\rho_2, \mu_2, \gamma_2, p_{\infty,2})^T = (1\,\mathrm{kg\,m^{-3}}, 1.725 \times 10^{-5}\,\mathrm{Pa\,s}, 1.4, 0\,\mathrm{Pa})^T$ and in the helium to $(\rho_1, \mu_1, \gamma_1, p_{\infty,1})^T = (0.138\,\mathrm{kg\,m^{-3}}, 1.868 \times 10^{-5}\,\mathrm{Pa\,s}, 1.667, 0\,\mathrm{Pa})^T$. The reference parameters used for non-dimensionalization are $\rho_{\mathrm{ref}} = \rho_2$, $x_{\mathrm{ref}} = D$ and $u_{\mathrm{ref}} = 376.64\,\mathrm{m\,s^{-1}}$. The domain is discretized into a base mesh of 200×100 equidistant mesh cells and a maximum of 3 levels of AMR with $\Delta p_{\mathrm{AMR}} = 0.57$.

The instantaneous volume fraction fields shown in Fig. 35 shows the advection and deformation behavior of the helium column over time, wherein the helium column initially gets compressed into a bean-like structure and at later stages a jet of air can be observed penetrating the bubble. This is even more apparent when looking at the 0.5 iso-contours of volume fraction shown in Fig. 36, where the penetrating jet creates a thin structure at the downstream location, which eventually ruptures due to insufficient mesh resolution to resolve the structure.

Figure 37 shows the idealized numerical schlieren images using $k = k_1 = k_2 = 25$ (see Eq. (4.51)) layered alongside the instantaneous AMR mesh colored by refinement level. The schlieren images show that the incident shock partially reflects and transmits through the column and plays the fundamental role in the deformation of the interface. The interface deformation behavior and shock patterns are comparable to the experiments [56] and other computational works [133, 74, 28, 78], although



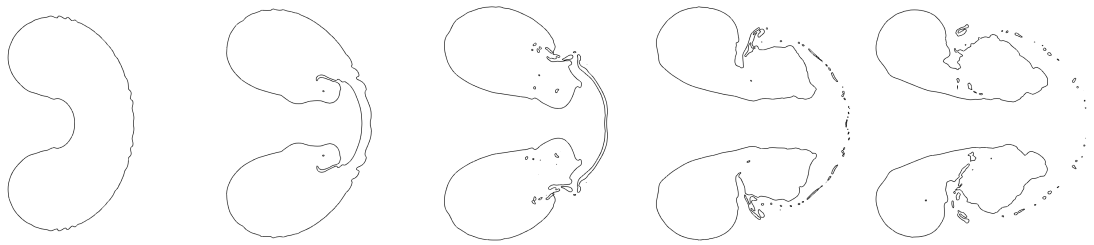
(a) $t = 0.02$ **(b)** $t = 0.04$

(c) $t = 0.06$ **(d)** $t = 0.08$

(e) $t = 0.12$ **(f)** $t = 0.16$

Figure 35: Volume fraction field for the Ma = 1.22 shock in air impacting a helium column at various instances in time

it should be noted that in the experiments mixing of the fluids occur, unlike the present work, where the fluids are assumed to be immiscible. The instantaneous AMR mesh shows the ability to resolve localized regions of shock and helium, while



(a) $t = 0.10$ **(b)** $t = 0.16$ **(c)** $t = 0.20$ **(d)** $t = 0.26$ **(e)** $t = 0.30$

Figure 36: Interface deformation time history depicted by 0.5 iso-contours of volume fraction for the Ma = 1.22 shock in air impacting a helium column

coarsening aggressively outwards. The highest refinement level provides an equivalent resolution of 1600×800 cells on an equidistant Cartesian mesh at the fraction of the computational cost.

Finally to verify that the solution to this case converges under mesh refinement, a Grid Convergence Index (GCI) analysis [134] of the jet position (Fig. 38) of the deformed interface at time $t = 0.2$ is performed, and is shown in Table 13. Simulations are performed on the base mesh with 1, 2, and 3 levels of maximum refinement and the jet position on the 0.5 iso-surface of volume fraction is computed. The observed order is estimated to be between second and third-order from the GCI analysis. The Richardson extrapolated value of the jet position at the finest mesh refinement level is estimated to be $0.258126 \pm 0.000387\%$, which is within the error bounds of the Richardson extrapolated value computed using a sharp interface method in [78], providing additional solution verification.

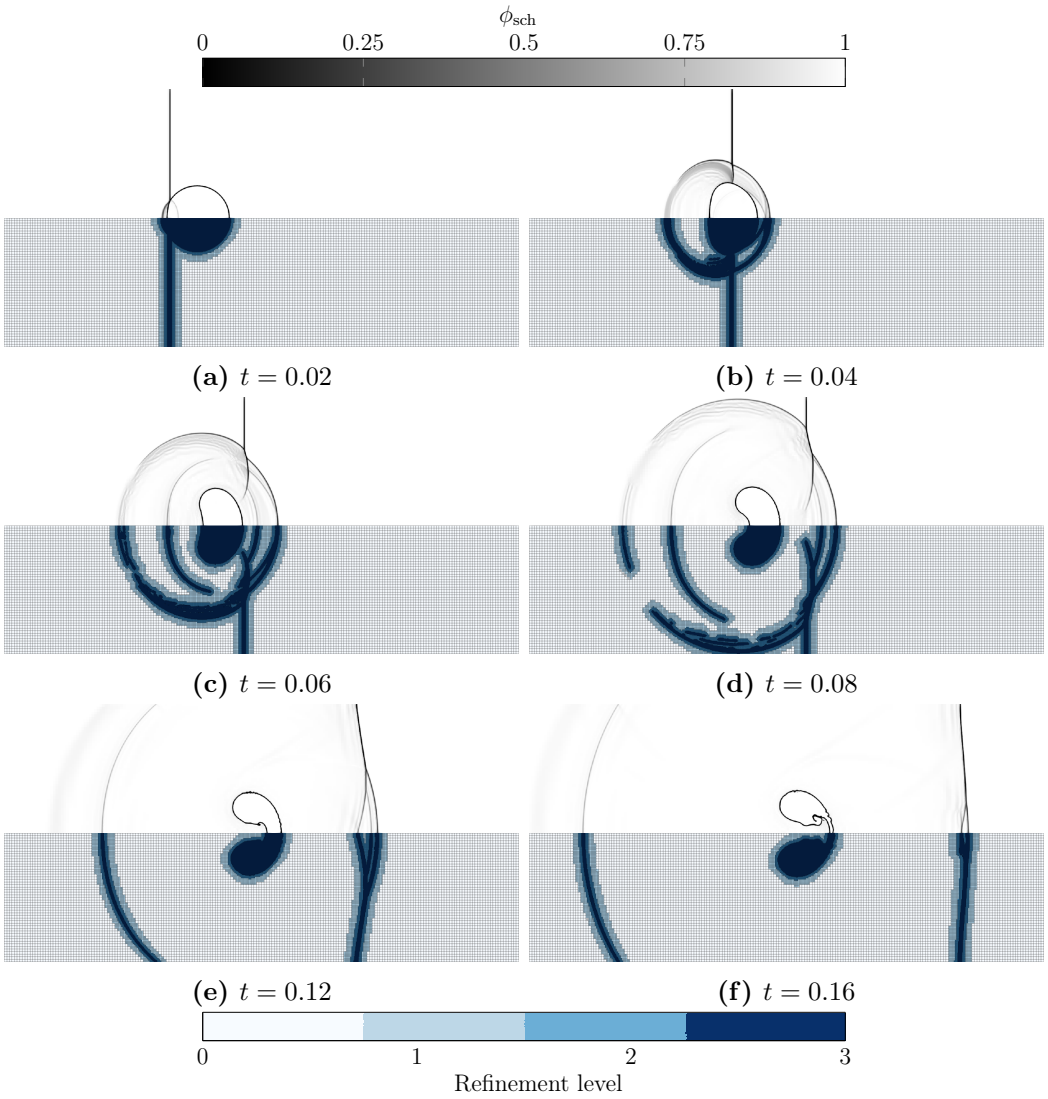


Figure 37: Idealized numerical schlieren images (upper half) and adaptive mesh refinement (lower half) for the Ma = 1.22 shock in air impacting a helium column at various instances in time

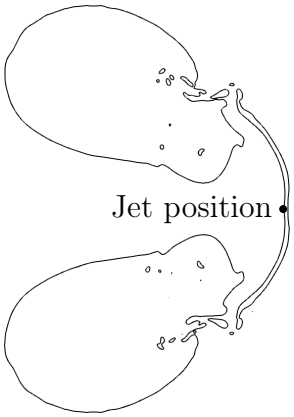


Figure 38: Position of interest considered for the GCI analysis of the shock in air impacting helium column test case

Table 13: Grid convergence index (GCI) analysis for the Ma = 1.22 shock in air impacting a helium column at $t = 0.2$

Levels of AMR	Interface position x	Observed order p	Richardson extrapolate x_0	GCI_{12} [%]	GCI_{23} [%]	Asymptotic range of convergence
1	0.2563	-	-	-	-	-
2	0.2577	-	-	-	-	-
3	0.2580	2.2538	0.258126	3.8×10^{-4}	1.8×10^{-3}	0.9988

4.5.4.2 Ma = 1.3 Shock in Air Impacting a Water Column

To study the performance of the implemented method in the context of stiff gas/liquid systems, the two-dimensional interaction of a Ma = 1.3 shock in air impacting a water column based on the experiments of Igra, Ogawa, and Takayama [70] is considered. A water column of diameter $D = 6.4\,\text{mm}$ is placed in a rectangular two-dimensional domain of size $[-0.016\,\text{m}, 0.032\,\text{m}] \times [-0.016\,\text{m}, 0.016\,\text{m}]$ with its center

at the origin. A planar shock of Mach number Ma = 1.3 under atmospheric pressure (101 325 Pa) is placed at $x = -0.003\,84$ m moving left to right. The initial condition in the pre-shock air is set to $(\rho_2, \mu_2, p_2)^T = (1.204\,\mathrm{kg\,m^{-3}}, 1.86 \times 10^{-5}\,\mathrm{Pa\,s}, 101\,325\,\mathrm{Pa})^T$ and in water to $(\rho_1, \mu_1, p_1)^T = (998.21\,\mathrm{kg\,m^{-3}}, 10.02 \times 10^{-4}\,\mathrm{Pa\,s}, p_2 + \sigma\kappa_0)^T$, where the initial curvature $\kappa_0 = 2/D$ and the surface tension coefficient σ is set corresponding to a Weber number of 3690 using the post-shock air and column diameter. The SG-EOS parameters are given by $(\gamma_2, p_{\infty,2})^T = (1.4, 0\,\mathrm{Pa})^T$ and $(\gamma_1, p_{\infty,1})^T = (6.12, 3.43 \times 10^8\,\mathrm{Pa})^T$. The reference parameters used for non-dimensionalization are $\rho_{\mathrm{ref}} = \rho_2$, $x_{\mathrm{ref}} = D$ and $u_{\mathrm{ref}} = 343.25\,\mathrm{m\,s^{-1}}$. The domain is discretized into a base mesh of 150×100 equidistant mesh cells and a maximum of 3 levels of AMR with $\Delta p_{\mathrm{AMR}} = 0.575$.

The idealized numerical schlieren images using $k_1 = k_2 = 5000$ layered with instantaneous AMR mesh colored by refinement level at various instances in time are shown in Figs. 39 and 40. In contrast to the helium column case examined in Section 4.5.4.1, the water column does not undergo significant deformation due to the impact of the incident shock wave, but instead starts to show signs of deformation at later times once the incident shock has completely passed. The instantaneous volume fraction fields shown in Fig. 41 reaffirm this behavior with the initial circular shape of the water column remaining mostly undisturbed until the later stages of the simulation. Figure 39 also shows the ability of the AMR with the chosen Δp_{AMR} to capture the shocks appropriately thereby providing an equivalent resolution of

1200×800 equidistant mesh cells, and coarsening aggressively otherwise. It should be noted that the small particle-like shedding observed in the numerical schlieren[8] is an artifact of the visualization equation used, Eq. (4.51), and is not an indication of breakup as confirmed in the volume fraction field shown in Fig. 41.

As discussed earlier, the column deformation is not primarily driven by the transmission or reflection of the shock, but instead is initially due to induced shear and at later stages, vortices downstream of the bubble (see Fig. 42) creating small-scale disturbances slowly over time. The solution presented in this work is in good qualitative agreement with experiments [71, 70] and the contribution of vortices towards the column deformation in similar air-water systems theorized by Engel [36] was also observed in [109, 45].

To verify that the solution converges with mesh refinement, a GCI analysis is performed on the center of mass position $\boldsymbol{x}_{\text{COM}}$ estimated following [109],

$$\boldsymbol{x}_{\text{COM}} = \frac{\int \rho_1 \alpha_1 \boldsymbol{x} \, d\mathcal{V}}{\int \rho_1 \alpha_1 \, d\mathcal{V}} \tag{4.52}$$

For the purposes of the analysis, only the $x-$component of the position vector is considered. Simulations are performed on the base mesh with 1, 2, and 3 levels of maximum refinement to a solution time of $t = 15$. Table 14 shows the results of the GCI analysis, wherein roughly between first and second order of convergence was

[8] This post-processing artifact was also observed in the work of Garrick, Hagen, and Regele [45].

observed for the four meshes considered. Furthermore, the analysis estimates the center of mass $x-$position to be 0.051221 using Richardson extrapolation with error bounds of $\pm 0.005475\%$ based on the finest resolution. Finally, the mesh refinement study also passes the check for an asymptotic range of convergence value of roughly one.

Table 14: Grid convergence index (GCI) analysis for the Ma = 1.3 shock in air impacting a water column at $t = 15$

Levels of AMR	Center of mass x	Observed order p	Richardson extrapolate x_0	GCI_{12} [%]	GCI_{23} [%]	Asymptotic range of convergence
0	0.0580	-	-	-	-	-
1	0.0539	-	-	-	-	-
2	0.0520	1.1039	0.0503	0.0399	0.0827	1.0367
3	0.0515	1.7875	0.0512	0.0055	0.0187	1.0107

4.5.4.3 Ma = 6 Shock in Air Impacting Two Water Columns

Finally, the implemented method is tested with a similar problem but with increased complexity, wherein the interaction of a strong incident shock with two staggered water columns is studied. Following the test case proposed in [21], a strong shock of Mach number Ma = 6 is placed at $x = -0.004\,\text{m}$ moving left to right in a rectangular domain of size identical to that considered in Section 4.5.4.2. The first water column of diameter $D_1 = 6.4\,\text{mm}$ is placed with its center at the

origin while a second water column of diameter $D_2 = 5\,\mathrm{mm}$ is placed staggered with its center at $(7\,\mathrm{mm}, -4\,\mathrm{mm})$. The initial condition in the pre-shock air is set to $(\rho_2, \mu_2, p_2)^T = (1.018\,\mathrm{kg\,m^{-3}}, 2.06 \times 10^{-5}\,\mathrm{Pa\,s}, 101\,325\,\mathrm{Pa})^T$ and in water to $(\rho_1, \mu_1, p_1)^T = (975.54\,\mathrm{kg\,m^{-3}}, 3.84 \times 10^{-4}\,\mathrm{Pa\,s}, p_2 + \sigma\kappa_0)^T$ following [21], where the initial curvature $\kappa_0 = 2/D_i$ and the surface tension coefficient $\sigma = 0.64\,\mathrm{N\,m^{-1}}$. The SG-EOS parameters are given by $(\gamma_2, p_{\infty,2})^T = (1.4, 0\,\mathrm{Pa})^T$ and $(\gamma_1, p_{\infty,1})^T = (1.93, 1.14 \times 10^9\,\mathrm{Pa})^T$. The reference parameters used for non-dimensionalization are $\rho_{\mathrm{ref}} = \rho_2$, $x_{\mathrm{ref}} = D_1$ and $u_{\mathrm{ref}} = 373.29\,\mathrm{m\,s^{-1}}$. The domain is discretized into a base mesh of 150×100 equidistant mesh cells and a maximum of 3 levels of AMR with $\Delta p_{\mathrm{AMR}} = 29.17$.

Figure 43 shows the idealized numerical schlieren images using $k_1 = 50$ and $k_2 = 500$ layered with the instantaneous AMR mesh shaded by refinement level at various instances in time. The incident shock is observed to be moving faster in the air than in the water owing to its high Mach number, unlike the test case in Section 4.5.4.2 where the higher speed of sound in water produces the opposite effect. The interface deformation behavior, however, has a similar evolution as the previous test case with the reflection and transmission of the incident shock not immediately deforming the columns owing to their higher densities. It is observed that the second water column sees a non-planar shock, as the incident shock gets refracted off the first water column, thereby creating an overall complex flow structure. A bow shock is seen upstream of the columns, and the vorticity created downstream of the columns plays a major role in the interface deformation, similar to the previous case. Furthermore,

the AMR is once again able to provide high-resolution in regions of shock and liquid for the given Δp_{AMR} at reduced computational costs. Finally, similar artifacts as in the previous test case are observed in the numerical schlieren but are not indicative of any physical breakup mechanism or the presence of phase interfaces.

The instantaneous volume fraction fields shown in Fig. 44 reaffirm this behavior as the number of small scale structures are significantly lower than seen in Fig. 43. The interface deformation can also be seen in further detail in Figs. 44 and 45 as observable deformations in the water columns show up only at later stages. The 0.5 iso-surfaces of volume fraction in Fig. 45 also displays small scale surface corrugations which are made possible by the low-dissipation schemes used in this work. The numerical solution is in good qualitative agreement with the computational works performed in Chang and Liou [21] and Hu, Adams, and Iaccarino [64].

Finally, a GCI analysis is performed on the center of mass $x-$position using Eq. (4.52) on the first water column to verify mesh convergence. Table 15 provides the details of the analysis for a base mesh with 1, 2, and 3 levels of maximum refinement. Using Richardson extrapolation, the center of mass $x-$position is estimated to be $0.132433 \pm 0.0489\%$ on the finest mesh level considered. Furthermore, the GCI analysis indicates roughly first order of convergence based on the three finest meshes.

Table 15: Grid convergence index (GCI) analysis for the Ma = 6 shock in air impacting two water columns at $t = 1.4$

Levels of AMR	Center of mass x	Observed order p	Richardson extrapolate x_0	GCI_{12} [%]	GCI_{23} [%]	Asymptotic range of convergence
0	0.1650	-	-	-	-	-
1	0.1514	-	-	-	-	-
2	0.1426	0.6134	0.1258	0.1470	0.2117	1.0623
3	0.1378	0.9092	0.1324	0.0489	0.0887	1.0343

4.6 Conclusions

An extension to a conservative finite volume method for the five-equation model is presented. Improvements include a hybrid reconstruction for the solution variables which resorts to a low-dissipation, high spatial order WENO-Z scheme in regions of pure phase and uses a THINC reconstruction scheme near the vicinity of the diffuse interface to maintain the thickness of the interface region. A fallback mechanism is proposed to combat the non-TVD property of the WENO-Z reconstruction, wherein local spatial accuracy is sacrificed to guarantee physicality of the solution variables. An extension to the height function method for estimating interface curvature is proposed to account for the thickness of the interface region. A redistribution scheme is used to propagate the interface curvature to the entirety of the interface region. Finally, a novel, cell-based, unstructured AMR framework is used to provide higher resolution in regions of interest.

The approach is used to simulate a number of test cases involving strong shocks, high density ratios and surface tension effects, obtaining good agreement with theoretical results, as well as experimental and computational observations. The results indicate that the high-fidelity extensions show improved performance over other works which used similar numerical approaches.

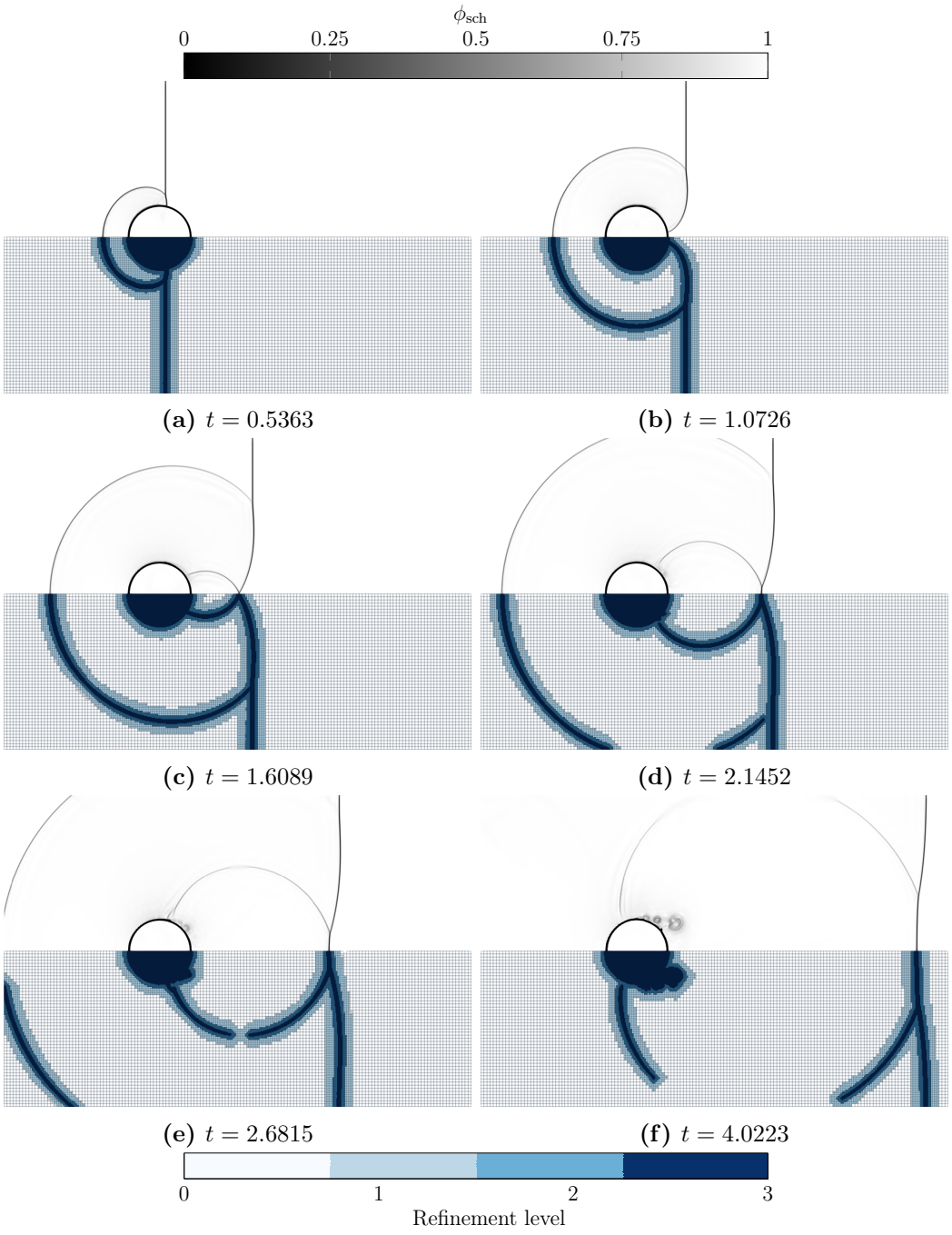


Figure 39: Idealized numerical schelieren images (upper half) and adaptive mesh refinement (lower half) for the Ma = 1.3 shock in air impacting a water column at various instances in time

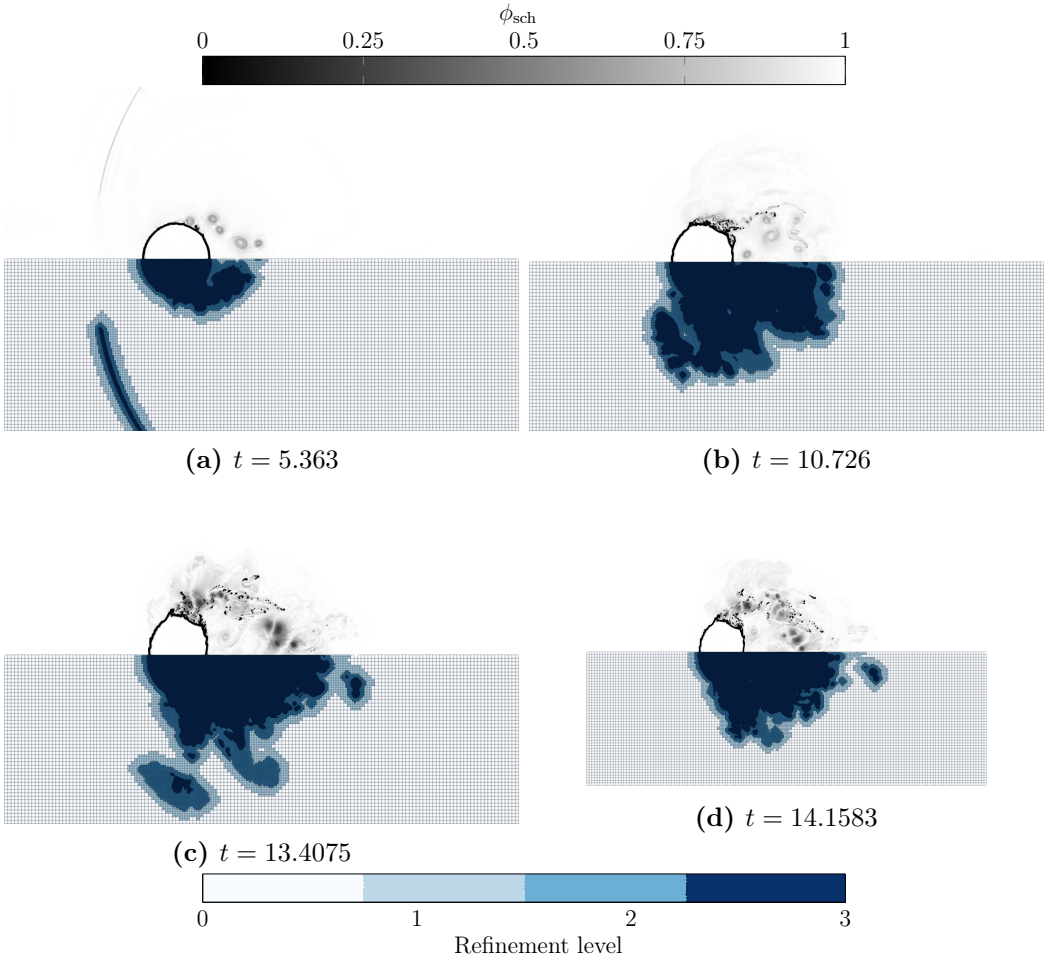
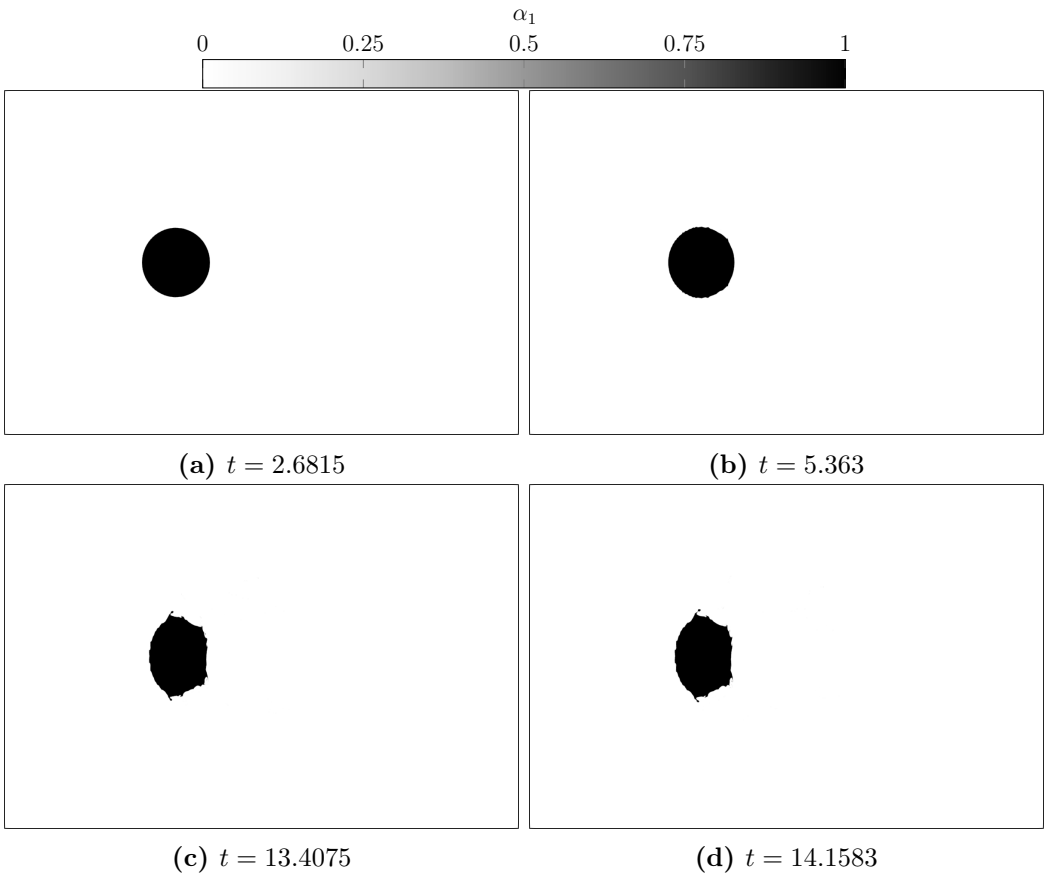


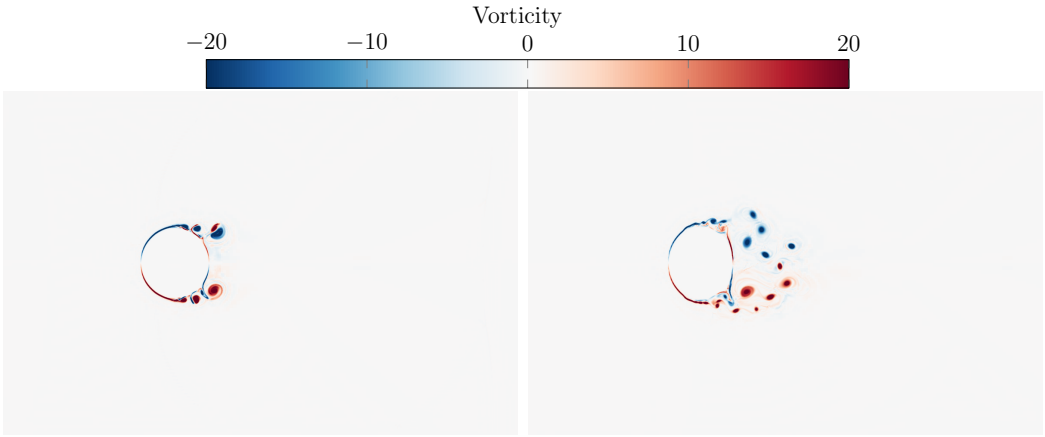
Figure 40: Idealized numerical schelieren images (upper half) and adaptive mesh refinement (lower half) for the Ma = 1.3 shock in air impacting a water column at various instances in time (continued)



(a) $t = 2.6815$ **(b)** $t = 5.363$

(c) $t = 13.4075$ **(d)** $t = 14.1583$

Figure 41: Volume fraction field for the Ma = 1.3 shock in air impacting a water column at various instances in time



(a) $t = 4.0223$ **(b)** $t = 8.0445$

(c) $t = 10.726$ **(d)** $t = 13.4075$

Figure 42: Out-of-plane vorticity component for the Ma = 1.3 shock in air impacting a water column at various instances in time

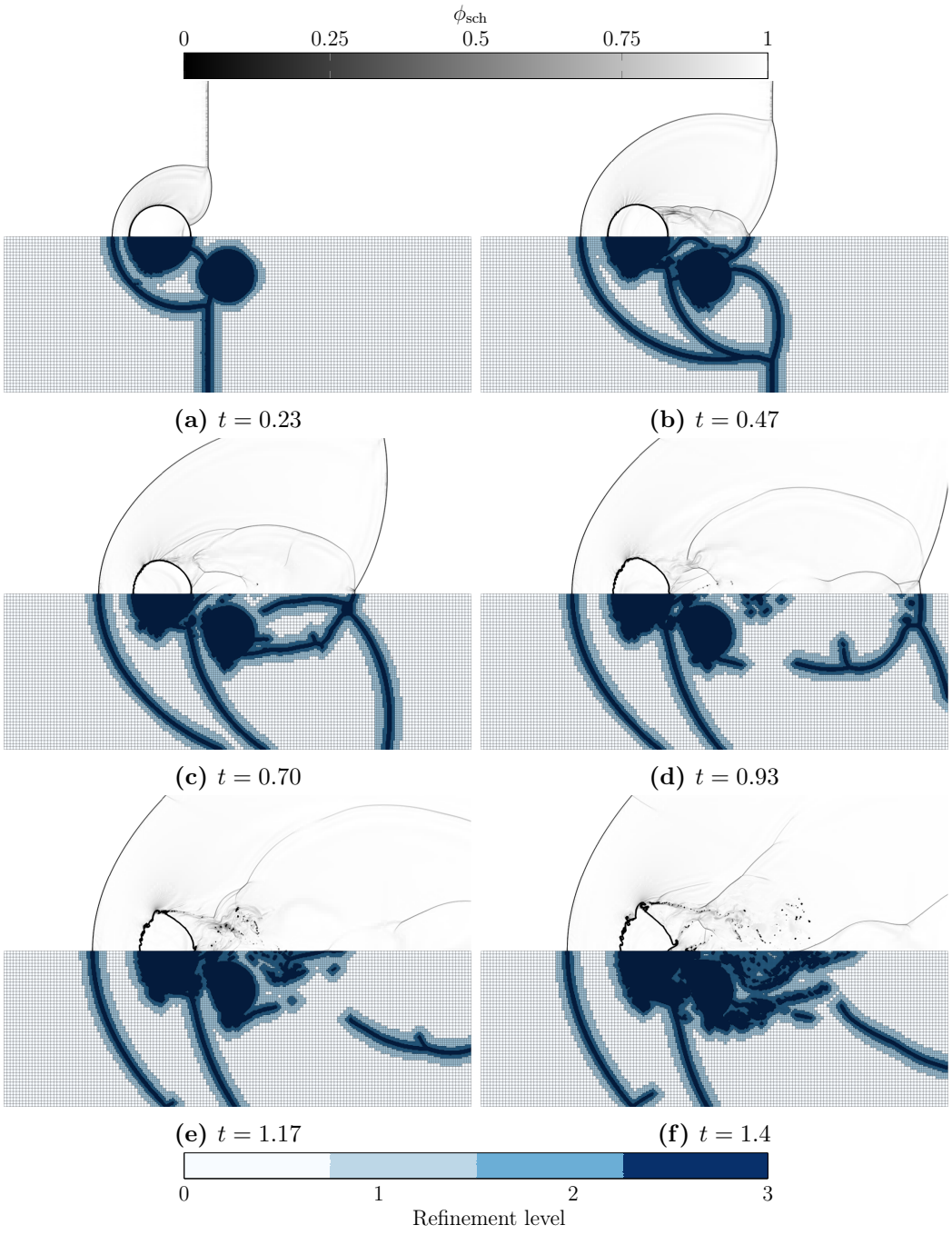
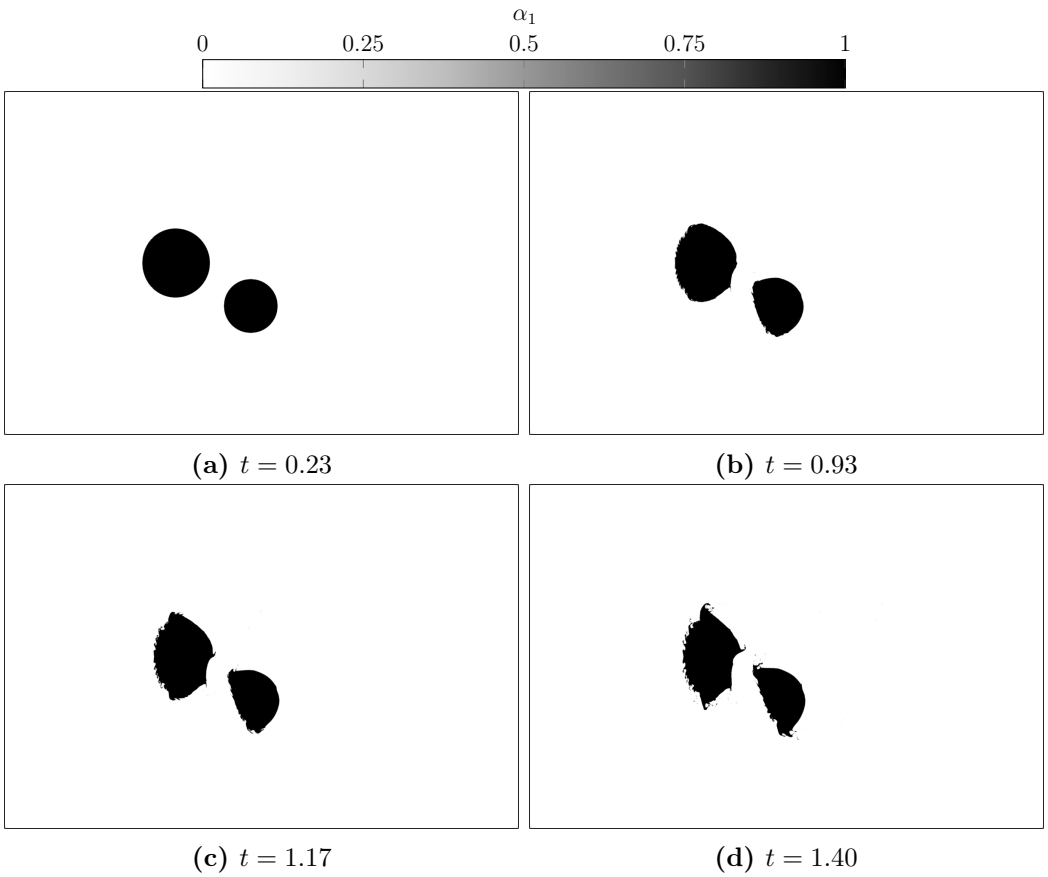


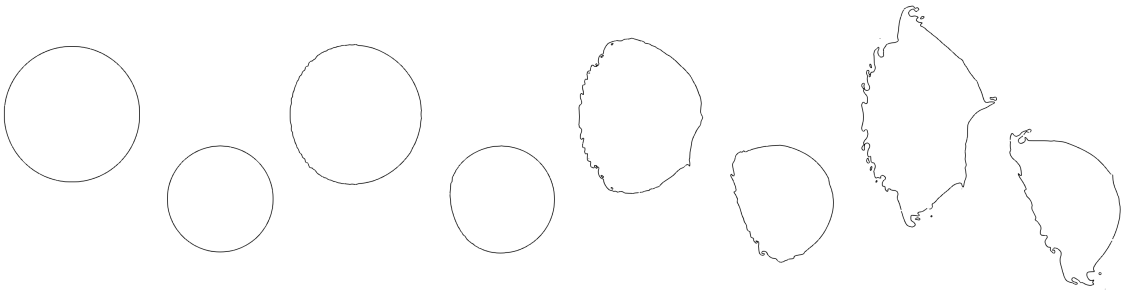
Figure 43: Idealized numerical schelieren images (upper half) and adaptive mesh refinement (lower half) for the Ma = 6 shock in air impacting two water columns at various instances in time



(a) $t = 0.23$ **(b)** $t = 0.93$

(c) $t = 1.17$ **(d)** $t = 1.40$

Figure 44: Volume fraction field for the Ma = 6 shock in air impacting two water columns at various instances in time



(a) $t = 0$ **(b)** $t = 0.47$ **(c)** $t = 0.93$ **(d)** $t = 1.40$

Figure 45: Interface deformation time history depicted by 0.5 iso-contours of volume fraction for the Ma = 6 shock in air impacting two water columns

Chapter 5

PRIMARY ATOMIZATION OF LIQUID JET IN A SUPERSONIC CROSSFLOW

In this chapter, the application test case of the primary atomization of a liquid jet in supersonic crossflow is considered. The simulation is performed with the intention of providing a preliminary study of the qualitative features of the flow.

5.1 Case Conditions

The simulation conditions for the application case are based on the experiments performed at Wright-Patterson Air Force Base [95, 98, 96]. A liquid jet is injected into a supersonic crossflow of gas at Ma = 1.94 through an injector with orifice diameter of $D = 0.51\,\text{mm}$. Based on experimental measurements, the total pressure and total temperature of the crossflow are 206 kPa and 533 K respectively. For the liquid, properties of water at room temperature are considered $(\rho_1, \mu_1)^T = (998\,\text{kg m}^{-3},\ 2.67 \times 10^{-3}\,\text{Pa s})^T$. The corresponding values in the free stream of air are $(\rho_2, \mu_2)^T = (0.3\,\text{kg m}^{-3},\ 1.73 \times 10^{-3}\,\text{Pa s})^T$. The momentum flux ratio, $q = \rho_1 u_1^2 / \rho_2 u_2^2$ is maintained at 13.64, following the experiments. The surface tension coefficient is taken as $0.07\,\text{N m}^{-1}$. The SG-EOS parameters for the fluids are $(\gamma_1, p_{\infty,1})^T = (4.4,\ 6 \times 10^8\,\text{Pa})^T$ and $(\gamma_2, p_{\infty,2})^T = (1.4,\ 0\,\text{Pa})^T$. For the simulation, the reference parameters used for non-dimensionalization are $\rho_{\text{ref}} = \rho_2$, $x_{\text{ref}} = D$ and $u_{\text{ref}} = 349.53\,\text{m s}^{-1}$.

The computational domain extends in the crossflow direction from $x = [-10D, 30D]$, in the injector-wall normal direction from $y = [0D, 30D]$ and in the transverse direction from $z = [-10D, 10D]$. The injector is located at the origin of the domain. For the result discussed in this chapter, a static uniform mesh with equidistant mesh spacing of $\Delta = D/8$ is used. Simulations with higher resolution utilizing the adaptive mesh refinement capability discussed in Chapter 4, providing a minimum resolution for the liquid region of $\Delta = D/32$ will be performed in the future.

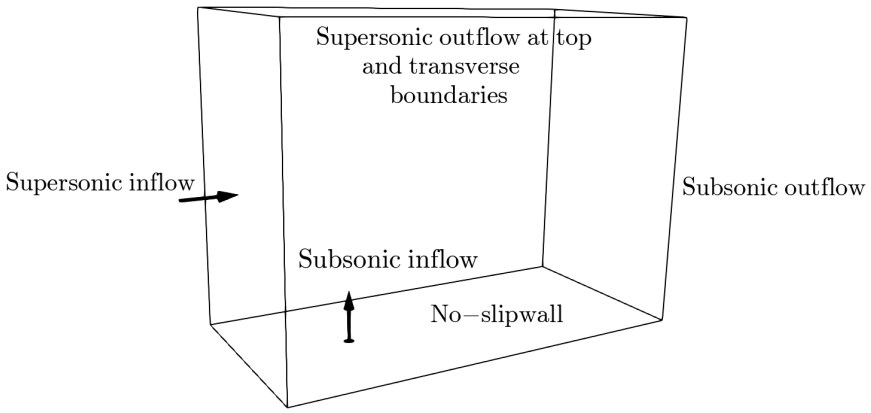


Figure 46: Schematic of computational domain with boundary conditions for the liquid jet in supersonic crossflow application case

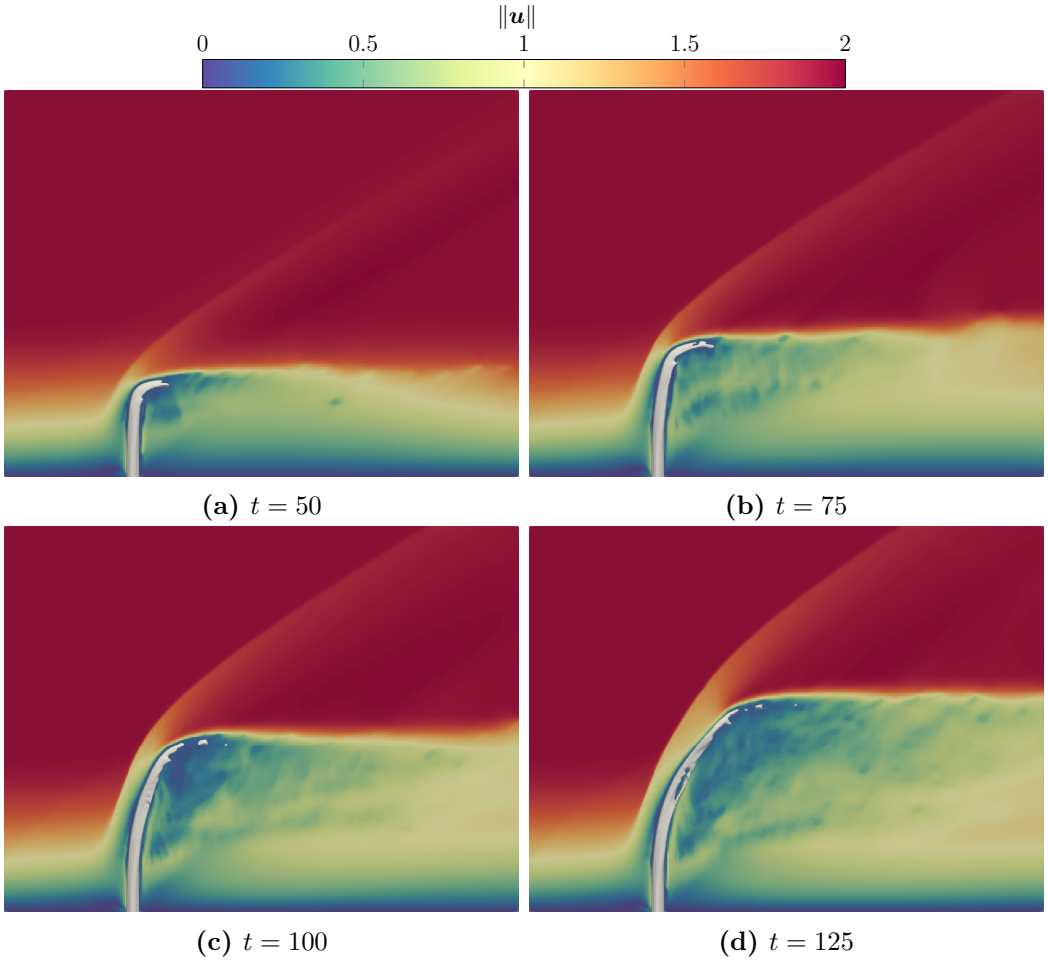
Figure 46 shows a representative domain with boundary conditions listed for all boundaries. A supersonic inflow boundary condition is specified at the lower end of the crossflow direction ($x-$direction), while the upper end has a subsonic outflow boundary condition specified. The lower wall in the $y-$direction contains both a

subsonic inflow for the liquid injector and a no-slip wall boundary condition elsewhere on that plane. Finally, in the transverse ($z-$direction) boundaries and the uppermost boundary, supersonic outflow boundary conditions are used.

Since the experiments measured a boundary layer thickness of 6.4 mm, a constant-in-time boundary layer velocity profile with the specified thickness is prescribed at the gas crossflow inlet. Furthermore, the entire domain is also initialized with the crossflow conditions before injecting the liquid. The Reynolds number of the liquid injector indicates a turbulent flow regime, and while turbulence modeling has not been considered as a part of the present research, a constant (in time) turbulent pipe flow velocity profile is used to model the inflow at the injector outlet.

5.2 Results and Discussion

The simulations are performed with a CFL of 0.3 and Fig. 47 shows the instantaneous snapshots of the initial stages of the simulation. As the liquid gets injected into the crossflow, a standing bow shock is gradually formed just upstream of the injector. Additionally, a separation shock can be observed emerging from the boundary layer that further interacts with the bow shock. Small droplets can be observed in the flow and they can be attributed to insufficient mesh resolution, rather than a physical breakup mechanism.



(a) $t = 50$ **(b)** $t = 75$

(c) $t = 100$ **(d)** $t = 125$

Figure 47: Instantaneous snapshots of the 0.5 iso-surface of volume fraction along with velocity magnitude displayed on the mid-plane along the transverse direction

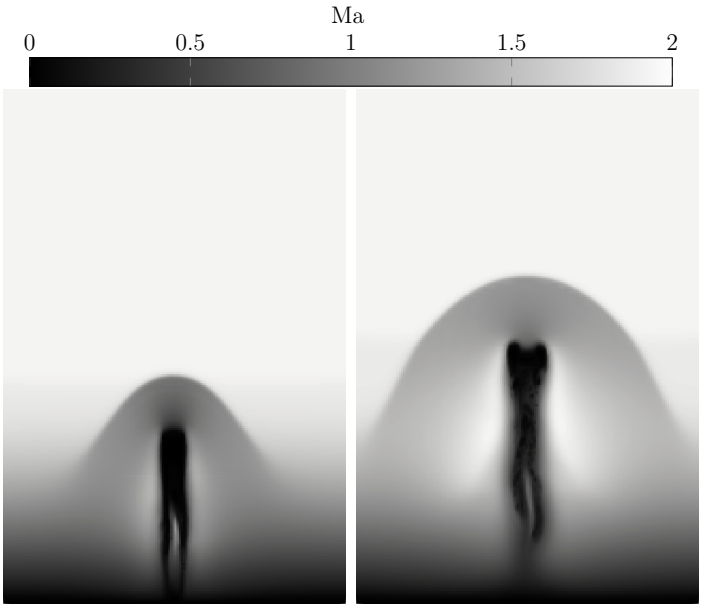
Inspection of the Mach numbers along stream-wise planes as shown in Fig. 48 reveal the three-dimensional nature of the bow shock, which is narrow near the injector and spreads wider moving further downstream. Several Mach reflections can also be noticed due to the presence of boundaries, but do not interfere with the

liquid due to the chosen size of the computational domain. Finally, there is also an indication of a pair of counter-rotating vortices aligned in the wall normal direction when looking at the Mach number.

This can be verified by looking at the stream-wise component of the vorticity vector as shown in Fig. 49, where a large vortex pair can be seen inducing strong upward flow at the center of the transverse domain. There is some indication of smaller pairs of vortices inducing downward flow away from the center, but the present simulation lacks sufficient detail due to the low mesh resolution. On a qualitative basis, the overall flow behavior is similar to that observed in [95, 98, 45] and shows a promising initial result of the developed method to be used for detailed analysis in the future.

5.3 Conclusions

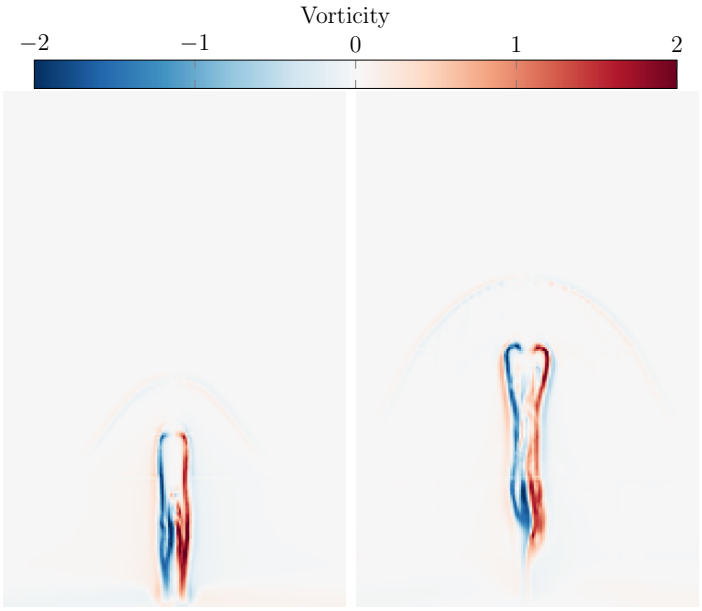
A preliminary study of the application case of a liquid jet in supersonic crossflow with the developed method has been performed. Qualitative analysis indicates the observed results are in agreement with the expected behavior, based on experiments and results in literature. Further detailed studies are warranted to obtain quantitative data and to validate the physics captured with experimental data. The use of AMR



(a) $x/D = 1$ **(b)** $x/D = 5$

(c) $x/D = 10$ **(d)** $x/D = 20$

Figure 48: Mach number on planes along the stream-wise direction at $t = 130$



(a) $x/D = 1$ **(b)** $x/D = 5$

(c) $x/D = 10$ **(d)** $x/D = 20$

Figure 49: Stream-wise component of vorticity on planes along the stream-wise direction at $t = 130$

will aid in performing high-resolution simulations at feasible computational costs. Moreover, the effect of turbulent and/or transient boundary conditions on this case proves to be another avenue of future research.

Chapter 6

CONCLUSIONS

The objective of this dissertation was to develop numerical methods capable of accurately simulating multiphase compressible flows, with a focus on primary atomization. Such flows involve many physical aspects such as topological changes of the phase interface, interactions with shocks, rarefaction waves and contact discontinuities. Additional complexity is introduced due to a jump in the equation of state required to provide a complete thermodynamic description of the fluids considered. Moreover, effects of surface tension and viscosity have been shown to be of importance for atomization processes in relevant literature.

Two different computational approaches were developed during the course of this research work. In Chapter 2, the initial efforts towards addressing the aforementioned challenges is described, wherein a novel sharp interface method was developed in a finite volume framework. This method coupled the single-phase compressible Navier-Stokes equations with a geometric volume-of-fluid interface tracking method using an in-cell reconstruction method, allowing for individual fluid quantities to be constructed from an averaged state. Furthermore, the method avoided the use of computational stencils across the phase interface and avoided the creation of cut-cells, both being major drawbacks of typical sharp interface methods. Surface tension effects were enforced directly into the reconstruction algorithm by using jump conditions and thereby foregoing any need for a surface tension model. The resulting method

behaves like a traditional interface tracking method, where the individual fluids are separated by geometric means, but has the ease of topology change of traditional interface capturing methods. The developed method was verified to capture all the incorporated physics accurately and demonstrated good performance for cases involving the interaction of shocks and immiscible interfaces.

The developed method however, was only applicable in a robust manner for flows involving two gases. Chapter 3 detailed the extensions required to improve the developed method to account for a jump in equation of state to be able to simulate gas-liquid flows. The interaction of generated waves with the phase interface was taken into account in an attempt to model the underlying physical nature. Numerical test cases indicate that the extension resulted in good performance for one-dimensional flows, but is unable to handle multi-dimensional flows in a robust manner. The additional steps required to make the extension truly multi-dimensional was outlined, but is non-trivial and creates additional complexity to the algorithm.

The second computational approach described in Chapter 4 provided an extension to the work by Coralic and Colonius [28] and Garrick, Hagen, and Regele [45], wherein a diffuse interface method is used to solve the multicomponent compressible Navier-Stokes equations in a finite volume framework. Low-dissipation, higher-order schemes were used to improve the overall accuracy of the method, while an extension to the standard height function method provided a mesh convergent approach for estimating interface curvature. Furthermore, the use of a novel, unstructured adaptive mesh refinement (AMR) framework aided in providing higher spatial resolution to

regions of shocks and liquid, while reducing spatial resolution elsewhere. Numerical tests demonstrated the ability of this approach to simulate gas-liquid flows in an accurate and efficient manner. Finally in Chapter 5, the developed method was used to perform a preliminary simulation of the primary atomization of liquid jet in Ma = 1.94 supersonic crossflow. The results of this simulation showed qualitative agreement to the overall behavior observed in experiments and paves a path forward to more detailed simulations.

Based on the outcomes and findings of this dissertation, the following recommendations can be made for future research. Firstly, a detailed simulation of the liquid jet in supersonic crossflow case using the demonstrated AMR capabilities in this work and providing validation against the experiments is warranted. Additionally, to more accurately represent the liquid injector used in the experiments, transient, turbulent inflow conditions need to be provided. This can be achieved either by using transient profiles from existing databases, or by modeling the flow in the injector geometry during the simulation. Moreover, the use of low-dissipation schemes in this work make the transition to turbulent simulations viable, and hence extending the current work to perform large eddy simulations (LES) is recommended. An alternative to the low-dissipation schemes could be the use of entropy conservative and entropy-stable schemes which are central schemes by construction and reduce numerical dissipation further. Another avenue for improvement is in the equation of state used. The stiffened-gas equation of state used in this work has been widely shown by many researchers [137, 21, 138, 41, 144, 45] to induce unphysical negative

pressures and is generally attributed to the lack of a cavitation model. More recently, new equations of state have been developed that have proven be an improvement over the stiffened-gas equation of state, such as the Nobel-Abel stiffened-gas equation of state [86]. Finally, the next step is the process of secondary atomization and hence, coupling of the current procedure with a secondary atomization model by means of droplet identification and droplet transfer is recommended.

REFERENCES

[1] Rémi Abgrall. "How to prevent pressure oscillations in multicomponent flow calculations: A quasi conservative approach". In: *J. Comput. Phys.* 125.1 (Apr. 1996), pp. 150–160.

[2] Rémi Abgrall and Smadar Karni. "Computations of Compressible Multifluids". In: *J. Comput. Phys.* 169.2 (May 2001), pp. 594–623.

[3] Grégoire Allaire, Sébastien Clerc, and Samuel Kokh. "A Five-Equation Model for the Simulation of Interfaces between Compressible Fluids". In: *Journal of Computational Physics* 181.2 (Sept. 2002), pp. 577–616.

[4] Marco Arienti and Marios C. Soteriou. "Time-resolved proper orthogonal decomposition of liquid jet dynamics". In: *Physics of Fluids* 21.11 (Nov. 2009). Publisher: American Institute of Physics, p. 112104.

[5] T. Aslam, S. Luo, and H. Zhao. "A static PDE Approach for MultiDimensional Extrapolation Using Fast Sweeping Methods". In: *SIAM Journal on Scientific Computing* 36.6 (Jan. 2014), A2907–A2928.

[6] M R Baer and J W Nunziato. "A two-phase mixture theory for the deflagration-to-detonation transition (ddt) in reactive granular materials". In: *International Journal of Multiphase Flow* 12.6 (Nov. 1986), pp. 861–889.

[7] Carlos Alberto Ballesteros et al. "A Parallel Adaptive Mesh Refinement Library for Cartesian Meshes". eng; English. Doctoral dissertation. Tempe: Arizona State University, 2019.

[8] Dinshaw S. Balsara and Chi-Wang Shu. "Monotonicity Preserving Weighted Essentially Non-oscillatory Schemes with Increasingly High Order of Accuracy". In: *Journal of Computational Physics* 160.2 (May 2000), pp. 405–452.

[9] P. Batten et al. "On the Choice of Wavespeeds for the HLLC Riemann Solver". In: *SIAM Journal on Scientific Computing* 18.6 (Nov. 1997). Publisher: Society for Industrial and Applied Mathematics, pp. 1553–1570.

[10] A. Ben-Yakar, M.G. Mungal, and R.K. Hanson. "Time evolution and mixing characteristics of hydrogen and ethylene transverse jets in supersonic crossflows". In: *Physics of Fluids* 18 (2006), p. 026101.

[11] Steven J. Beresh et al. "Crossplane Velocimetry of a Transverse Supersonic Jet in a Transonic Crossflow". In: *AIAA Journal* 44.12 (2006), pp. 3051–3061.

[12] Marsha J Berger and Joseph Oliger. "Adaptive mesh refinement for hyperbolic partial differential equations". In: *Journal of Computational Physics* 53.3 (1984), pp. 484–512.

[13] Wurigen Bo and John W Grove. "A volume of fluid method based ghost fluid method for compressible multi-fluid flows". In: *Comput. Fluids* 90 (2014), pp. 113–122.

[14] Wurigen Bo and John W. Grove. "A volume of fluid method based ghost fluid method for compressible multi-fluid flows". In: *Computers & Fluids* 90 (Feb. 2014), pp. 113–122.

[15] J Boles et al. "Hybrid large-eddy simulation/reynolds-averaged Navier–Stokes simulations of sonic injection into Mach 2 crossflow". In: *49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*. AIAA Meeting Papers, Jan. 2011.

[16] Rafael Borges et al. "An improved weighted essentially non-oscillatory scheme for hyperbolic conservation laws". In: *Journal of Computational Physics* 227.6 (Mar. 2008), pp. 3191–3211.

[17] J. U Brackbill, D. B Kothe, and C Zemach. "A continuum method for modeling surface tension". In: *Journal of Computational Physics* 100.2 (June 1992), pp. 335–354.

[18] Carsten Burstedde, Lucas C. Wilcox, and Omar Ghattas. "`p4est`: Scalable Algorithms for Parallel Adaptive Mesh Refinement on Forests of Octrees". In: *SIAM Journal on Scientific Computing* 33.3 (2011), pp. 1103–1133.

[19] Xiaochuan Chai and Krishnan Mahesh. "Simulations of High Speed Turbulent Jets in Crossflows". In: *49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*. AIAA Meeting Papers, Jan. 2011.

[20] Chih-Hao Chang, Xiaolong Deng, and Theo G. Theofanous. "Direct numerical simulation of interfacial instabilities: A consistent, conservative, all-speed, sharp-interface method". In: *Journal of Computational Physics* 242 (2013), pp. 946–990.

[21] Chih-Hao Chang and Meng-Sing Liou. "A robust and accurate approach to computing compressible multiphase flow: Stratified flow model and AUSM+-up scheme". In: *Journal of Computational Physics* 225.1 (July 2007), pp. 840–873.

[22] Clarence F. Chenault, Philip S. Beran, and Rodney D.W. Bowersox. "Numerical Investigation of Supersonic Injection Using a Reynolds-Stress Turbulence Model". In: *AIAA Journal* 37.10 (1999), pp. 1257–1269.

[23] Alexandre Chiapolino, Richard Saurel, and Boniface Nkonga. "Sharpening diffuse interfaces with compressible fluids on unstructured meshes". In: *Journal of Computational Physics* 340 (2017), pp. 389–417.

[24] J. P. Cocchi, R. Saurel, and J. C. Loraud. "Treatment of interface problems with Godunov-type schemes". In: *Shock Waves* 5.6 (May 1996), pp. 347–357.

[25] Jean-Pierre Cocchi and Richard Saurel. "A Riemann Problem Based Method for the Resolution of Compressible Multimaterial Flows". In: *J. Comput. Phys.* 137.2 (Nov. 1997), pp. 265–298.

[26] Jean-Pierre Cocchi and Richard Saurel. "A Riemann Problem Based Method for the Resolution of Compressible Multimaterial Flows". In: *Journal of Computational Physics* 137.2 (Nov. 1997), pp. 265–298.

[27] P Colella, H M Glaz, and R E Ferguson. *Multifluid algorithms for Eulerian finite- difference methods*. Tech. rep. Apr. 1997.

[28] Vedran Coralic and Tim Colonius. "Finite-volume WENO scheme for viscous compressible multicomponent flows". In: *Journal of Computational Physics* 274 (Oct. 2014), pp. 95–121.

[29] Sharen J. Cummins, Marianne M. Francois, and Douglas B. Kothe. "Estimating curvature from volume fractions". In: *Computers & Structures*. Frontier of Multi-Phase Flow Analysis and Fluid-Structure 83.6 (Feb. 2005), pp. 425–434.

[30] Edward T. Curran. "Scramjet Engines: The First Forty Years". In: *J. Propul. Power* 17.6 (2001), pp. 1138–1148.

[31] F. Daude et al. "Numerical experiments using a HLLC-type scheme with ALE formulation for compressible two-phase flows five-equation models with phase transition". In: *Computers and Fluids* 94 (2014), pp. 112–138.

[32] F. Daude et al. "Numerical experiments using a HLLC-type scheme with ALE formulation for compressible two-phase flows five-equation models with phase transition". In: *Computers & Fluids* 94 (2014), pp. 112–138.

[33] Xi Deng et al. "High fidelity discontinuity-resolving reconstruction for compressible multiphase flows with moving interfaces". In: *Journal of Computational Physics* 371 (Oct. 2018), pp. 945–966.

[34] Anshu Dubey et al. "A survey of high level frameworks in block-structured adaptive mesh refinement packages". In: *Journal of Parallel and Distributed Computing* 74.12 (2014), pp. 3217–3227.

[35] B Einfeldt et al. "On Godunov-type methods near low densities". In: *Journal of Computational Physics* 92.2 (1991), pp. 273–295.

[36] O.G. Engel. "Fragmentation of waterdrops in the zone behind an air shock". In: *Journal of Research of the National Bureau of Standards* 60.3 (Mar. 1958), p. 245.

[37] Lord Rayleigh O. M. F.R.S. "VIII. On the pressure developed in a liquid during the collapse of a spherical cavity". In: *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 34.200 (Aug. 1917), pp. 94–98.

[38] Charbel Farhat, Jean-Frédéric Gerbeau, and Arthur Rallu. "FIVER: A finite volume method based on exact two-phase Riemann problems and sparse grids for multi-material flows with large density jumps". In: *J. Comput. Phys.* 231.19 (Aug. 2012), pp. 6360–6379.

[39] Charbel Farhat, Jean-Frédéric Gerbeau, and Arthur Rallu. "FIVER: A finite volume method based on exact two-phase Riemann problems and sparse grids for multi-material flows with large density jumps". In: *Journal of Computational Physics* 231.19 (2012), pp. 6360–6379.

[40] S. Fechter and C.-D. Munz. "A discontinuous Galerkin-based sharp-interface method to simulate three-dimensional compressible two-phase flow". In: *International Journal for Numerical Methods in Fluids* 78 (2015), pp. 413–435.

[41] S. Fechter and C.-D. Munz. "A discontinuous Galerkin-based sharp-interface method to simulate three-dimensional compressible two-phase flow". In: *International Journal for Numerical Methods in Fluids* 78.7 (2015), pp. 413–435.

[42] R Fedkiw et al. "A non-oscillatory Eulerian approach to interfaces in multimaterial flows (the ghost fluid method)". In: *J. Comput. Phys.* 152 (1999), pp. 457–492.

[43] Ronald P Fedkiw et al. "A Non-oscillatory Eulerian Approach to Interfaces in Multimaterial Flows (the Ghost Fluid Method)". In: *Journal of Computational Physics* 152.2 (July 1999), pp. 457–492.

[44] Message Passing Interface Forum. *MPI: A Message-Passing Interface Standard.* Tech. rep. USA, 1994.

[45] Daniel P. Garrick, Wyatt A. Hagen, and Jonathan D. Regele. "An interface capturing scheme for modeling atomization in compressible flows". In: *Journal of Computational Physics* 344 (Sept. 2017), pp. 260–280.

[46] Daniel P. Garrick, Mark Owkes, and Jonathan D. Regele. "A finite-volume HLLC-based scheme for compressible interfacial flows with surface tension". In: *Journal of Computational Physics* 339 (June 2017), pp. 46–67.

[47] P Glaister. "An approximate linearised riemann solver for the three-dimensional euler equations for real gases using operator splitting". In: *J. Comput. Phys.* 77.2 (1988), pp. 361–383.

[48] J Glimm, D Marchesin, and O McBryan. "Subgrid resolution of fluid discontinuities, II". In: *J. Comput. Phys.* 37.3 (Oct. 1980), pp. 336–354.

[49] J Glimm, D Marchesin, and O McBryan. "Subgrid resolution of fluid discontinuities, II". In: *Journal of Computational Physics* 37.3 (Oct. 1980), pp. 336–354.

[50] J. Glimm et al. "Conservative Front Tracking with Improved Accuracy". In: *SIAM Journal on Numerical Analysis* 41.5 (Jan. 2003), pp. 1926–1947.

[51] James Glimm et al. "Conservative Front Tracking with Improved Accuracy". In: *SIAM J. Numer. Anal.* 41.5 (Jan. 2003), pp. 1926–1947.

[52] Sergei Konstantinovich Godunov. "A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics". In: *Mat. Sb. (N.S.)* 47(89).3 (1959), pp. 271–306.

[53] John W. Grove et al. "Quantitative theory of Richtmyer-Meshkov instability". In: *Phys. Rev. Lett.* 71 (21 Nov. 1993), pp. 3473–3476.

[54] Mark R. Gruber, Abdollah S. Nejad, and J.C. Dutton. *An Experimental Investigation of Transverse Injection from Circular and Elliptical Nozzles into a Supersonic Crossflow.* Tech. rep. WL-TR-96-2102. Ohio: WRIGHT LAB WRIGHT-PATTERSON AFB, 1996.

[55] Brian T.N. Gunney, Andrew M. Wissink, and David A. Hysom. "Parallel clustering algorithms for structured AMR". In: *Journal of Parallel and Distributed Computing* 66.11 (2006), pp. 1419–1430.

[56] J.-F. Haas and B. Sturtevant. "Interaction of weak shock waves with cylindrical and spherical gas inhomogeneities". en. In: *Journal of Fluid Mechanics* 181 (Aug. 1987), pp. 41–76.

[57] F.H. Harlow and A.A. Amsden. *FLUID DYNAMICS. A LASL Monograph.* Technical Report LA-4700. Los Alamos Scientific Laboratory, Jan. 1971, p. 115.

[58] Ami Harten and James M Hyman. "Self adjusting grid methods for one-dimensional hyperbolic conservation laws". In: *J. Comput. Phys.* 50.2 (1983), pp. 235–269.

[59] M Herrmann. "Detailed Numerical Simulations of the Primary Atomization of a Turbulent Liquid Jet in Crossflow". In: *Journal of Engineering for Gas Turbines and Power* 132.6 (2010), pp. 061506–061510.

[60] M Herrmann, P Moin, and S Abarzhi. "Nonlinear evolution of the Richtmyer-Meshkov instability". In: *J. Fluid Mech.* 612 (2008), pp. 311–338.

[61] M. Herrmann. "A balanced force refined level set grid method for two-phase flows on unstructured flow solver grids". In: *Journal of Computational Physics* 227.4 (Feb. 2008), pp. 2674–2706.

[62] X Y Hu, N A Adams, and G Iaccarino. "On the HLLC Riemann solver for interface interaction in compressible multi-fluid flow". In: *J. Comput. Phys.* 228.17 (Sept. 2009), pp. 6572–6589.

[63] X Y Hu et al. "A conservative interface method for compressible flows". In: *J. Comput. Phys.* 219.2 (Dec. 2006), pp. 553–578.

[64] X. Y. Hu, N. A. Adams, and G. Iaccarino. "On the HLLC Riemann solver for interface interaction in compressible multi-fluid flow". In: *Journal of Computational Physics* 228.17 (Sept. 2009), pp. 6572–6589.

[65] X. Y. Hu et al. "A conservative interface method for compressible flows". In: *Journal of Computational Physics* 219.2 (Dec. 2006), pp. 553–578.

[66] Wei Huang et al. "Investigation on the flameholding mechanisms in supersonic flows: backward-facing step and cavity flameholder". In: *Journal of Visualization* 14.1 (Feb. 2011), pp. 63–74.

[67] Wei Huang et al. "Investigation on the flameholding mechanisms in supersonic flows: backward-facing step and cavity flameholder". In: *Journal of Visualization* 14.1 (2011), pp. 63–74.

[68] Wei Huang et al. "Research status of key techniques for shock-induced combustion ramjet (shcramjet) engine". In: *Science in China Series E: Technological Sciences* 53.1 (Jan. 2010), pp. 220–226.

[69] Wei Huang et al. "Research status of key techniques for shock-induced combustion ramjet (shcramjet) engine". In: *Science in China Series E: Technological Sciences* 53.1 (2010), pp. 220–226.

[70] D. Igra, T. Ogawa, and K. Takayama. "A PARAMETRIC STUDY OF WATER COLUMN DEFORMATION RESULTING FROM SHOCK WAVE LOADING". English. In: *Atomization and Sprays* 12.5&6 (2002).

[71] D. Igra and K. Takayama. "Numerical simulation of shock wave interaction with a water column". en. In: *Shock Waves* 11.3 (Sept. 2001), pp. 219–228.

[72] Matthew Jemison, Mark Sussman, and Marco Arienti. "Compressible, multiphase semi-implicit method with moment of fluid interface representation". In: *J. Comput. Phys.* 279 (Dec. 2014), pp. 182–217.

[73] Guang-Shan Jiang and Chi-Wang Shu. "Efficient Implementation of Weighted ENO Schemes". In: *Journal of Computational Physics* 126 (1996), pp. 202–228.

[74] Eric Johnsen and Tim Colonius. "Implementation of WENO schemes in compressible multicomponent flow problems". In: *Journal of Computational Physics* 219.2 (Dec. 2006), pp. 715–732.

[75] Eric Johnsen and Tim Colonius. "Shock-induced collapse of a gas bubble in shockwave lithotripsy". In: *The Journal of the Acoustical Society of America* 124.4 (Oct. 2008). Publisher: Acoustical Society of America, pp. 2011–2020.

[76] Samet Y. Kadioglu et al. "A second order primitive preconditioner for solving all speed multi-phase flows". In: *Journal of Computational Physics* 209.2 (2005), pp. 477–503.

[77] James Russell Kamm. *An Exact, Compressible One-Dimensional Riemann Solver for General, Convex Equations of State*. Tech. rep. LA-UR–15-21616. Los Alamos, NM (United States): Los Alamos National Lab. (LANL), Mar. 2015.

[78] Karthik Kannan, Dominic Kedelty, and Marcus Herrmann. "An in-cell reconstruction finite volume method for flows of compressible immiscible fluids". In: *Journal of Computational Physics* 373 (Nov. 2018), pp. 784–810.

[79] S. Kawai and S.K. Lele. "Dynamics and mixing of a sonic jet in a supersonic turbulent crossflow". In: *Center for Turbulence Research Annual Research Briefs* (2009), pp. 285–298.

[80] D. Ketcheson. "Highly Efficient Strong Stability-Preserving Runge–Kutta Methods with Low-Storage Implementations". In: *SIAM Journal on Scientific Computing* 30.4 (Jan. 2008), pp. 2113–2136.

[81] Keiichi Kitamura and Taku Nonomura. "Simple and robust HLLC extensions of two-fluid AUSM for multiphase flow computations". In: *Computers and Fluids* 100 (2014), pp. 321–335.

[82] V.P. Kolgan. "Application of the principle of minimizing the derivative to the construction of finite-difference schemes for computing discontinuous solutions of gas dynamics". In: *Journal of Computational Physics* 230.7 (2011), pp. 2384–2390.

[83] B. Koren et al. "Riemann-Problem and Level-Set Approaches for Homentropic Two-Fluid Flow Computations". In: *Journal of Computational Physics* 181.2 (2002), pp. 654–674.

[84] M. Kornfeld and L. Suvorov. "On the Destructive Action of Cavitation". In: *Journal of Applied Physics* 15.6 (June 1944). Publisher: American Institute of Physics, pp. 495–506.

[85] Sir Horace Lamb. *Hydrodynamics*. English. 6th Revised ed. edition. New York: Dover Publications, June 1945.

[86] Olivier Le Métayer and Richard Saurel. "The Noble-Abel Stiffened-Gas equation of state". In: *Physics of Fluids* 28.4 (Apr. 2016), p. 046102.

[87] Jinho Lee, Kuo-Cheng Lin, and Dean Eklund. "Challenges in Fuel Injection for High-Speed Propulsion Systems". In: *AIAA Journal* 53.6 (2015), pp. 1405–1423.

[88] Bram van Leer. "Towards the ultimate conservative difference scheme. V. A second-order sequel to Godunov's method". In: *Journal of Computational Physics* 32.1 (July 1979), pp. 101–136.

[89] Bram Van Leer. "Towards the ultimate conservative difference scheme. IV. A new approach to numerical convection". In: *J. Comput. Phys.* 23.3 (1977), pp. 276–299.

[90] Randall J LeVeque. "High Resolution Finite Volume Methods on Arbitrary Grids Via Wave Propagation". In: *J. Comput. Phys.* 78 (1988), pp. 36–63.

[91] Randall J. LeVeque. "A large Time Step Generalization of Godunov's Method for Systems of Conservation Laws". In: *SIAM J. Numer. Anal.* 22.6 (1985), pp. 1051–1073.

[92] Randall J. LeVeque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press, 2010.

[93] Randall J. Leveque. *Finite Volume Methods for Hyperbolic Problems*. English. 1 edition. Cambridge ; New York: Cambridge University Press, Aug. 2002.

[94] Randall J. LeVeque and Keh-Ming Shyue. "One-Dimensional Front Tracking Based on High Resolution Wave Propagation Methods". In: *SIAM J. Sci. Comput.* 16.2 (1995), pp. 348–377.

[95] Kuo-Cheng Lin. *Potential New Data Sets to Assist Numerical Simulations*. Tech. rep. Wright-Patterson Air Force Base: Air Force Research Laboratory, 2016.

[96] Kuo-Cheng Lin, Timothy Ombrello, and Campbell D. Carter. "Qualitative Study of Near-Field and Cross-Sectional Structures of Liquid Jets in Supersonic Crossflow". In: *2018 AIAA Aerospace Sciences Meeting*. American Institute of Aeronautics and Astronautics, 2018.

[97] Kuo-Cheng Lin et al. "Raman Scattering Measurements of Gaseous Ethylene Jets in a Mach 2 Supersonic Crossflow". In: *J. Propul. Power* 26.3 (2010), pp. 503–513.

[98] Kuo-Cheng Lin et al. "Structures and Temporal Evolution of Liquid Jets in Supersonic Crossflow". In: *55th AIAA Aerospace Sciences Meeting*. AIAA SciTech Forum. American Institute of Aeronautics and Astronautics, Jan. 2017.

[99] Xu-Dong Liu, Stanley Osher, and Tony Chan. "Weighted Essentially Non-oscillatory Schemes". en. In: *Journal of Computational Physics* 115.1 (Nov. 1994), pp. 200–212.

[100] T G Liu, B C Khoo, and K S Yeo. "Ghost fluid method for strong shock impacting on material interface". In: *J. Comput. Phys.* 190.2 (Sept. 2003), pp. 651–681.

[101] T.G. Liu, B.C. Khoo, and C.W. Wang. "The ghost fluid method for compressible gas–water simulation". In: *Journal of Computational Physics* 204.1 (2005), pp. 193–221.

[102] J. López et al. "An improved height function technique for computing interface curvature from volume fractions". In: *Computer Methods in Applied Mechanics and Engineering* 198.33 (July 2009), pp. 2555–2564.

[103] J. Luitjens and M. Berzins. "Scalable Parallel Regridding Algorithms for Block-Structured Adaptive Mesh Refinement". In: *Concurrency and Computation: Practice and Experience* 23.13 (Sept. 2011), pp. 1522–1537.

[104] Hong Luo, Joesph D. Baum, and Rainald Lohner. "On the computation of multi-material flows using ALE formulation". In: *Journal of Computational Physics* 194.1 (2004), pp. 304–328.

[105] Hong Luo, Joseph D Baum, and Rainald Löhner. "On the computation of multi-material flows using ALE formulation". In: *Journal of Computational Physics* 194.1 (2004), pp. 304–328.

[106] Krishnan Mahesh. "The Interaction of Jets with Crossflow". In: *Annual Review of Fluid Mechanics* 45.1 (2013), pp. 379–407.

[107] Sahand Majidi and Asghar Afshari. "An adaptive interface sharpening methodology for compressible multiphase flows". en. In: *Computers & Mathematics with Applications* 72.10 (Nov. 2016), pp. 2660–2684.

[108] Sahand Majidi and Asghar Afshari. "Supersonic Liquid Jets Into Quiescent Gaseous Media: An Adaptive Numerical Study". In: *Journal of Fluids Engineering* 138.1 (Jan. 2016), p. 011103.

[109] J. C. Meng and T. Colonius. "Numerical simulations of the early stages of high-speed droplet breakup". en. In: *Shock Waves* 25.4 (July 2015), pp. 399–414.

[110] G H Miller and P Colella. "A Conservative Three-Dimensional Eulerian Method for Coupled Solid–Fluid Shock Capturing". In: *J. Comput. Phys.* 183.1 (Nov. 2002), pp. 26–82.

[111] Suman Muppidi and Krishnan Mahesh. "Direct numerical simulation of round turbulent jets in crossflow". In: *Journal of Fluid Mechanics* 574 (2007), pp. 59–84.

[112] Suman Muppidi and Krishnan Mahesh. "Optimization of pulsed jets in crossflow". In: *Journal of Fluid Mechanics* 653 (2010), pp. 365–390.

[113] Angelo Murrone and Hervé Guillard. "A five equation reduced model for compressible two phase flow problems". In: *J. Comput. Phys.* 202.2 (Jan. 2005), pp. 664–698.

[114] D de Niem, E Kührt, and U Motschmann. "A volume-of-fluid method for simulation of compressible axisymmetric multi-material flow". In: *Computer Physics Communications* 176.3 (Feb. 2007), pp. 170–190.

[115] D. [de Niem], E. Kührt, and U. Motschmann. "A volume-of-fluid method for simulation of compressible axisymmetric multi-material flow". In: *Computer Physics Communications* 176.3 (2007), pp. 170–190.

[116] W Noh and P Woodward. "SLIC (Simple Line Interface Calculation)". In: *Proc. 5th Int. Conf. Fluid Dyn.* Ed. by A van de Vooren and P Zandbergen. Berlin: Springer-Verlag, 1976, pp. 330–340.

[117] R R Nourgaliev, T N Dinh, and T G Theofanous. "Adaptive characteristics-based matching for compressible multifluid dynamics". In: *J. Comput. Phys.* 213.2 (2006), pp. 500–529.

[118] R. R. Nourgaliev et al. "The Characteristics-Based Matching Method (CBM) for High-Speed Fluid-Fluid Flows". In: *Computational Fluid Dynamics 2004*. Ed. by Clinton Groth and David W. Zingg. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 607–612.

[119] Mark Owkes and Olivier Desjardins. "A computational framework for conservative, three-dimensional, unsplit, geometric transport with application to the volume-of-fluid (VOF) method". In: *J. Comput. Phys.* 270.Supplement C (2014), pp. 587–612.

[120] Mark Owkes and Olivier Desjardins. "A mesh-decoupled height function method for computing interface curvature". In: *Journal of Computational Physics* 281 (Jan. 2015), pp. 285–300.

[121] D Papamoschou and D.G. Hubbard. "Visual observations of supersonic transverse jets". In: *Exp. Fluids* 14 (1993), pp. 468–476.

[122] George Shu Heng Pau et al. "An adaptive mesh refinement algorithm for compressible two-phase flow in porous media". en. In: *Computational Geosciences* 16.3 (June 2012), pp. 577–592.

[123] Guillaume Perigaud and Richard Saurel. "A compressible flow model with capillary effects". In: *Journal of Computational Physics* 209.1 (Oct. 2005), pp. 139–178.

[124] D. Peterson and G. Candler. "Simulations of mixing for normal and low-angled injection into a supersonic crossflow". In: *AIAA Journal* 49.12 (2011), pp. 2792–2804.

[125] A. Philipp et al. "Interaction of lithotripter-generated shock waves with air bubbles". In: *The Journal of the Acoustical Society of America* 93.5 (May 1993). Publisher: Acoustical Society of America, pp. 2496–2509.

[126] J E Pilliod. "An analysis of piecewise linear interface reconstruction algorithms for volume-of-fluid methods". MA thesis. University of California Davis, 1992.

[127] James Edward Pilliod Jr and Elbridge Gerry Puckett. "Second-order accurate volume-of-fluid algorithms for tracking material interfaces". In: *J. Comput. Phys.* 199.2 (Sept. 2004), pp. 465–502.

[128] S Popinet and S Zaleski. "A Front-Tracking Algorithm for Accurate Representation of Surface Tension". In: *Int. J. Numer. Meth. Fluids* 30 (1999), pp. 775–793.

[129] Stéphane Popinet. "Gerris: a tree-based adaptive solver for the incompressible Euler equations in complex geometries". In: *Journal of Computational Physics* 190.2 (2003), pp. 572–600.

[130] Stephane Popinet. "An accurate adaptive solver for surface-tension-driven interfacial flows". In: *J. Comput. Phys.* 228.16 (2009), pp. 5838–5866.

[131] Stephane Popinet. "Numerical Models of Surface Tension". In: *Annu. Rev. Fluid Mech.* 50.1 (Jan. 2018), pp. 49–75.

[132] Bruce D. Pratte and W. Douglas Baines. "Profiles of the Round Turbulent Jet in A Cross Flow". In: *Journal of the Hydraulics Division* 93.6 (1967), pp. 53–64.

[133] James J. Quirk and S. Karni. "On the dynamics of a shock–bubble interaction". en. In: *Journal of Fluid Mechanics* 318 (July 1996), pp. 129–163.

[134] P. J. Roache. “Perspective: A Method for Uniform Reporting of Grid Refinement Studies”. In: *Journal of Fluids Engineering* 116.3 (Sept. 1994), pp. 405–413.

[135] P.J. Roache. “Perspective: A Method for Uniform Reporting of Grid Refinement Studies”. In: *J. Fluids Eng* 116 (1994), pp. 405–413.

[136] P.L Roe. “Approximate Riemann solvers, parameter vectors, and difference schemes”. In: *J. Comput. Phys.* 43.2 (1981), pp. 357–372.

[137] Richard Saurel and Rémi Abgrall. “A Multiphase Godunov Method for Compressible Multifluid and Multiphase Flows”. In: *J. Comput. Phys.* 150.2 (Apr. 1999), pp. 425–467.

[138] Richard Saurel and O Lemetayer. “A multiphase model for compressible flows with interfaces, shocks, detonation waves and cavitation”. In: *Journal of Fluid Mechanics* 431 (Mar. 2001), pp. 239–271.

[139] Richard Saurel and Carlos Pantano. “Diffuse-Interface Capturing Methods for Compressible Two-Phase Flows”. In: *Annual Review of Fluid Mechanics* 50 (2018), pp. 105–130.

[140] Ruben Scardovelli and Stéphane Zaleski. “Analytical Relations Connecting Linear Interfaces and Volume Fractions in Rectangular Grids”. In: *J. Comput. Phys.* 164.1 (Oct. 2000), pp. 228–237.

[141] Ruben Scardovelli and Stéphane Zaleski. “Direct Numerical Simulation of Free-Surface and Interfacial Flow”. In: *Annu. Rev. Fluid Mech.* 31 (1999), pp. 567–603.

[142] Kevin Schmidmayer et al. “ECOGEN: An open-source tool for multiphase, compressible, multiphysics flows”. en. In: *Computer Physics Communications* 251 (June 2020), p. 107093.

[143] H Schmidt and R Klein. “A Generalized Level-Set/in-Cell-Reconstruction Approach for Accelerating Turbulent Premixed Flames”. In: *Combust. Theor. Model.* 7 (2003), pp. 243–267.

[144] S. Sembian et al. “Plane shock wave interaction with a cylindrical water column”. In: *Physics of Fluids* 28.5 (May 2016). Publisher: American Institute of Physics, p. 056102.

[145] J. Sethian. “Fast Marching Methods”. In: *SIAM Review* 41.2 (Jan. 1999), pp. 199–235.

[146] Jerry W. Shan and Paul E. Dimotakis. “Reynolds-number effects and anisotropy in transverse-jet mixing”. In: *Journal of Fluid Mechanics* 566 (2006), pp. 47–96.

[147] S Shin et al. “Accurate representation of surface tension using the level contour reconstruction method”. In: *J. Comput. Phys.* 203 (2005), pp. 493–516.

[148] C. Shu. “High Order Weighted Essentially Nonoscillatory Schemes for Convection Dominated Problems”. In: *SIAM Rev.* 51.1 (Feb. 2009), pp. 82–126.

[149] Chi-Wang Shu and Stanley Osher. “Efficient implementation of essentially non-oscillatory shock-capturing schemes”. In: *Journal of Computational Physics* 77.2 (Aug. 1988), pp. 439–471.

[150] Ratnesh K. Shukla, Carlos Pantano, and Jonathan B. Freund. “An interface capturing method for the simulation of multi-phase compressible flows”. In: *Journal of Computational Physics* 229.19 (Sept. 2010), pp. 7411–7439.

[151] Keh-Ming Shyue. “A Volume-of-fluid Type Algorithm for Compressible Two-phase Flows”. In: *Hyperbolic Problems: Theory, Numerics, Applications*. Basel: Birkhäuser Basel, 1999, pp. 895–904.

[152] Keh-Ming Shyue. “A wave-propagation based volume tracking method for compressible multicomponent flow in two space dimensions”. In: *Journal of Computational Physics* 215 (2006), pp. 219–244.

[153] Keh-Ming Shyue. “An Efficient Shock-Capturing Algorithm for Compressible Multicomponent Problems”. In: *J. Comput. Phys.* 142.1 (May 1998), pp. 208–242.

[154] Keh-Ming Shyue. “Front tracking methods based on wave propagation”. PhD thesis. University of Washington: University of Washington, 1993.

[155] Keh-Ming Shyue and Feng Xiao. "An Eulerian interface sharpening algorithm for compressible two-phase flow: The algebraic THINC approach". In: *Journal of Computational Physics* 268 (July 2014), pp. 326–354.

[156] V Smiljanovski. "Ein numerisches Verfahren zur Berechnung schneller Vormischflammen und der Deflagrations-Detonations-Transition (DDT)". PhD thesis. RWTH Aachen, 1996.

[157] V. Smiljanovski, V. Moser, and R. Klein. "A capturing - tracking hybrid scheme for deflagration discontinuities". In: *Combust. Theor. Model.* 1 (1997), pp. 183–215.

[158] K. K. So, X. Y. Hu, and N. A. Adams. "Anti-diffusion interface sharpening technique for two-phase compressible flow simulations". In: *Journal of Computational Physics* 231.11 (June 2012), pp. 4304–4323.

[159] L.K. Su and M.G. Mungal. "Simultaneous measurements of scalar and velocity field evolution in turbulent crossflowing jets". In: *Journal of Fluid Mechanics* 513 (2004), pp. 1–45.

[160] William Sutherland. "LII. The viscosity of gases and molecular force". In: *Philosophical Magazine* 36.223 (1893), pp. 507–531.

[161] Hiroshi Terashima and Gretar Tryggvason. "A front-tracking/ghost-fluid method for fluid interfaces in compressible flows". In: *J. Comput. Phys.* 228.11 (June 2009), pp. 4012–4037.

[162] Tiankai Tu, D. R. O'Hallaron, and O. Ghattas. "Scalable Parallel Octree Meshing for TeraScale Applications". In: *SC '05: Proceedings of the 2005 ACM/IEEE Conference on Supercomputing*. Nov. 2005, pp. 4–4.

[163] V. A. Titarev and E. F. Toro. "Finite-volume WENO schemes for three-dimensional conservation laws". In: *Journal of Computational Physics* 201.1 (Nov. 2004), pp. 238–260.

[164] Arpit Tiwari, Jonathan B. Freund, and Carlos Pantano. "A diffuse interface model with immiscibility preservation". In: *Journal of Computational Physics* 252 (2013), pp. 290–309.

[165] Eleuterio F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction*. en. 3rd ed. Berlin Heidelberg: Springer-Verlag, 2009.

[166] D. J. Torres and J. U. Brackbill. "The Point-Set Method: Front-Tracking without Connectivity". In: *Journal of Computational Physics* 165.2 (Dec. 2000), pp. 620–644.

[167] Bing Wang and Houqian Xu. "A Method Based on Riemann Problem in Tracking Multi-Material Interface On Unstructured Moving Grids". In: *Engineering Applications of Computational Fluid Mechanics* 1.4 (2007), pp. 325–336.

[168] Bing Wang and Houqian Xu. "A method based on Riemann problem in tracking multi-material interface on unstructured moving grids". In: *Engineering Applications of Computational Fluid Mechanics* 1.4 (2007), pp. 325–336.

[169] C W Wang, T G Liu, and B C Khoo. "A Real Ghost Fluid Method for the Simulation of Multimedium Compressible Flow". In: *SIAM J. Sci. Comput.* 28.1 (Jan. 2006), pp. 278–302.

[170] Andrew B. Wardlaw. *Underwater Explosion Test Cases*. en. Tech. rep. NSWC-IHTR-2069. NAVAL SURFACE WARFARE CENTER INDIAN HEAD DIV MD, May 1998.

[171] Andrew B. Wardlaw and Hans U. Mair. "Spherical Solutions of an Underwater Explosion Bubble". In: *Shock and Vibration* 5 (Jan. 1900). Publisher: IOS Press, p. 690105.

[172] M. W. Williams, D. B. Kothe, and E. G. Puckett. "Accuracy and Convergence of Continuum Surface Tension Models". In: *Fluid Dynamics at Interfaces*. Univ. Press, 1998, pp. 294–305.

[173] F Xiao et al. "Large eddy simulation of liquid jet primary breakup in supersonic air crossflow". In: *International Journal of Multiphase Flow* 87 (Dec. 2016), pp. 229–240.

[174] F. Xiao, Y. Honma, and T. Kono. "A simple algebraic interface capturing scheme using hyperbolic tangent function". In: *International Journal for Numerical Methods in Fluids* 48.9 (2005), pp. 1023–1040.

[175] Feng Xiao, Satoshi Ii, and Chungang Chen. "Revisit to the THINC scheme: A simple algebraic VOF algorithm". In: *Journal of Computational Physics* 230.19 (Aug. 2011), pp. 7086–7092.

[176] Joseph Yang, Toshi Kubota, and Edward E. Zukoski. "Applications of shock-induced mixing to supersonic combustion". In: *AIAA Journal* 31.5 (May 1993). Publisher: American Institute of Aeronautics and Astronautics, pp. 854–862.

[177] David P Young et al. "A locally refined rectangular grid finite element method: Application to computational fluid dynamics and computational physics". In: *Journal of Computational Physics* 92.1 (1991), pp. 1–66.

[178] Qiang Zhang and Mary Jane Graham. "A numerical study of Richtmyer–Meshkov instability driven by cylindrical shocks". In: *Physics of Fluids* 10.4 (1998), pp. 974–992.

APPENDIX A

DERIVATION OF EQUATIONS FOR RECONSTRUCTION OF PRIMITIVE VARIABLES

The following outlines the derivation of the reconstruction equations for the primitive variables in the two fluids separated by the interface, Eq. (2.24).

The relations for the reconstructed fluid velocities from the cell averaged mixed states is trivial from the jump condition for velocity, Eq. (2.9a), resulting in $u_0 = u_1 = \frac{\overline{\rho u}}{\overline{\rho}}, v_0 = v_1 = \frac{\overline{\rho v}}{\overline{\rho}}, w_0 = w_1 = \frac{\overline{\rho w}}{\overline{\rho}}$. From the definition of the mixed state cell average energy, Eq. (2.23), we have:

$$\overline{\rho E} = \alpha \, (\rho E)_1 + (1-\alpha) \, (\rho E)_0$$

Using Eqs. (2.3) and (2.6),

$$\overline{\rho E} = \alpha \left(\frac{p_1}{\gamma_1 - 1} + \frac{1}{2} \rho_1 \left\| \boldsymbol{u} \right\|_1^2 \right) + (1-\alpha) \left(\frac{p_0}{\gamma_0 - 1} + \frac{1}{2} \rho_0 \left\| \boldsymbol{u} \right\|_0^2 \right)$$

with, $\left\| \boldsymbol{u} \right\|^2 = u^2 + v^2 + w^2$. Rearranging and combining,

$$p_0 = \frac{(\gamma_0 - 1)(\gamma_1 - 1)\left(\overline{\rho E} - \frac{1}{2}\rho \left\| \boldsymbol{u} \right\|^2\right) - \alpha p_1 (\gamma_0 - 1)}{(1-\alpha)(\gamma_1 - 1)}$$

Using the jump condition for pressure, Eq. (2.9b), $p_1 = p_0 + \sigma\kappa + [\![\boldsymbol{n}^T \cdot \overline{\overline{\tau}} \cdot \boldsymbol{n}]\!]_\Gamma$,

$$p_0 = \frac{(\gamma_0 - 1)(\gamma_1 - 1)\left(\overline{\rho E} - \frac{1}{2}\rho \left|\left| \boldsymbol{u} \right|\right|^2\right)}{\alpha(\gamma_0 - 1) + (1-\alpha)(\gamma_1 - 1)} - \frac{\alpha(\gamma_0 - 1)\left(\sigma\kappa + [\![\boldsymbol{n}^T \cdot \overline{\overline{\tau}} \cdot \boldsymbol{n}]\!]_\Gamma\right)}{\alpha(\gamma_0 - 1) + (1-\alpha)(\gamma_1 - 1)} \tag{A.1}$$

Similarly, applying the definition of the mixed state cell average density, Eq. (2.23),

$$\overline{\rho} = \alpha \, \rho_1 + (1-\alpha) \, \rho_0$$

and using Eqs. (2.8) and (2.9c) results in

$$\begin{aligned} T &= \frac{1}{\overline{\rho}} \left(\alpha \frac{p_1}{(\gamma_1 - 1) \, c_{v,1}} + (1-\alpha) \frac{p_0}{(\gamma_0 - 1) \, c_{v,0}} \right) \\ &= \frac{1}{\overline{\rho}} \left(\alpha \frac{p_0 + \sigma\kappa + [\![\boldsymbol{n}^T \cdot \overline{\overline{\tau}} \cdot \boldsymbol{n}]\!]_\Gamma}{(\gamma_1 - 1) \, c_{v,1}} + (1-\alpha) \frac{p_0}{(\gamma_0 - 1) \, c_{v,0}} \right) \end{aligned} \tag{A.2}$$

Substituting the expression for p_0 obtained previously in Eq. (A.1) results in an equation for $T = T_1 = T_2$,

$$\begin{aligned} &A_0 \, T + A_1 \mu_1 \, \boldsymbol{n}^T \cdot \left(\nabla \boldsymbol{u}_1 + (\nabla \boldsymbol{u}_1)^T - \frac{2}{3} (\nabla \boldsymbol{u}_1) \, \overline{\overline{I}} \right) \cdot \boldsymbol{n} \\ &- A_1 \mu_0 \, \boldsymbol{n}^T \cdot \left(\nabla \boldsymbol{u}_0 + (\nabla \boldsymbol{u}_0)^T - \frac{2}{3} (\nabla \boldsymbol{u}_0) \, \overline{\overline{I}} \right) \cdot \boldsymbol{n} - A_2 - A_3 = 0 \end{aligned}$$

with the coefficients A_0 through A_3 given in Eq. (2.26). Applying the viscosity model given in Eq. (2.5), the non-linear equation for T, Eq. (2.25) is recovered. To obtain the relation for individual densities, start with,

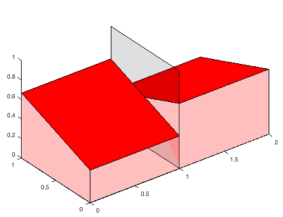
$$\overline{\rho} = \alpha \frac{p_1}{T\left(\gamma_1 - 1\right) c_{v,1}} + \left(1 - \alpha\right) \rho_0$$
$$\rho_0 = \frac{\overline{\rho} - \alpha\, p_1}{T\left(1 - \alpha\right)\left(\gamma_1 - 1\right) c_{v,1}}$$

Substituting the definition of T obtained previously in Eq. (A.2), it is possible to obtain the equation for ρ_0, see Eq. (2.24). The relation for ρ_1 can be obtained in a similar fashion starting from the definition of $\overline{\rho}$.

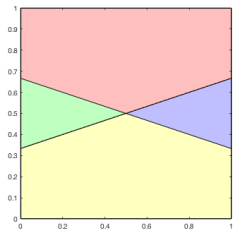
APPENDIX B

GEOMETRIC UN-SPLIT VOF SCHEME FOR INTERFACE WAVE TRANSPORT

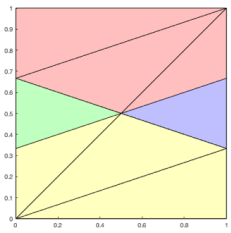
The motion of the interface wave is given by the solution to the volume fraction equation, Eq. (2.21). We solve it using an extension to the geometric framework proposed by Owkes and Desjardins [119]. The original framework developed for incompressible flows is an unsplit, conservative, and bounded method, which fluxes a complete quadrilateral cell face between two cells. To properly take into account the contribution of the interface wave, see Eqs. (2.31) and (2.33), we cannot flux the entire cell face as a whole as in [119], but instead need to flux separately the portions of the cell face that have either the same or different fluids on either side, resulting in the separate flux volumes $\Omega_{i,j}$ and corresponding volume fractions $\alpha_{i,j}$, with $i = 0, 1$ and $j = 0, 1$.



(a) Example of interface discontinuity at cell face due to PLIC reconstruction



(b) Example of wave contributions sorted by color



(c) Example of triangulated face sorted by color

Figure 50: Cell face areas for VoF flux volumes $\Omega_{0,0}$ (red), $\Omega_{1,1}$ (yellow), $\Omega_{0,1}$ (blue), and $\Omega_{1,0}$ (green).

As an example, Fig. 50a shows two adjacent cells with PLIC interface reconstructions. The depicted interface geometry results in cell face portions that have fluid 0 on both sides (red), fluid 1 on both sides (yellow), fluid 0 on the left and fluid 1 on the right side of the face (blue) and fluid 1 on the left side and fluid 0 on the right side (green), see Fig. 50b. Depending on the interface geometry, the individual cell face portions can be n-sided polygons with arbitrary orientation. Instead of fluxing these cell face polygons directly, we split them into triangles, see Fig. 50c, and flux these individually, resulting in a simplified, yet generally applicable fluxing algorithm for any cell face shape, as long as the shape can be triangulated.

The procedure for the triangulation is the following. The quadrilateral cell face is first split into two initial right-triangular faces where vertices that define that face are labeled in a clockwise direction with respect to the face normal direction. Using PLIC based reconstruction of the interface geometry [140], the equation for the interface plane in a cell based coordinate system is simply

$$\boldsymbol{n} \cdot \boldsymbol{x} = d\,, \tag{B.1}$$

where the interface normal n is calculated using ELVIRA [126, 127]. Using the above definition, the intersection of the interfaces on either side of the cell face with the cell face itself is easy to obtain, for a total of two possible intersections. The intersections are then used to split the cell face triangles into sub-triangles, by first calculating a signed distance $D_{\mathcal{P}}$ for each vertex $\mathcal{P}$ that defines the face triangle,

$$D_{\mathcal{P}} = \mathbf{n} \cdot (\boldsymbol{x}_{\mathcal{P}} - d|\mathbf{n}|)\,. \tag{B.2}$$

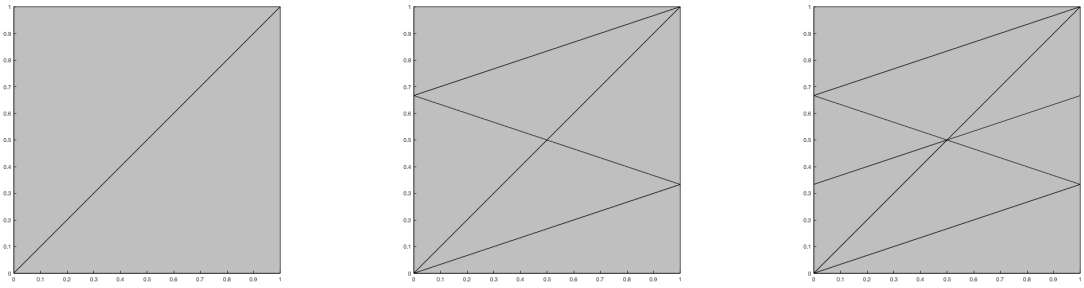
Based on the value of $D_{\mathcal{P}}$, four different cases can be identified with three leading to a splitting of the triangle into three sub-triangles based on which vertex is separated from the other two, and the fourth leaving the original triangle intact. Algorithm 3 summarizes the procedure, where $\mathbf{V}_{in}$ denotes the list of 3 vertices that define the triangle to be cut, and $\mathbf{V}_{out}$ denotes a list of 3 sub-triangles with 3 vertices each that define the result of the procedure.

Algorithm 3 Single Triangle Cut Function

input: $\mathbf{n}$, d, $\mathbf{V}_{in}$

$\mathbf{A} = \mathbf{V}_{in,1}$
$\mathbf{B} = \mathbf{V}_{in,2}$
$\mathbf{C} = \mathbf{V}_{in,3}$
Calculate D_A, D_B, D_C ▷ Calculate Signed Distances, Eq. (B.2)
CaseNumber $= \mathcal{H}(D_A) + 2\mathcal{H}(D_B) + 4\mathcal{H}(D_C)$ ▷ $\mathcal{H}$: Heaviside function
switch (CaseNumber)
    case 0 or 7 ▷ PLIC does not intersect triangle.
        $\mathbf{V}_{out,1} = \mathbf{V}_{in}$
        $\mathbf{V}_{out,2} = [-\mathbf{1}, -\mathbf{1}, -\mathbf{1}]$
        $\mathbf{V}_{out,2} = [-\mathbf{1}, -\mathbf{1}, -\mathbf{1}]$
        skipTriangles = [**false**, **true**, **true**]
    case 1 or 6 ▷ PLIC intersects **AB** and **AC**
        Calculate **E** ▷ Point intersection of PLIC with **AB**
        Calculate **F** ▷ Point intersection of PLIC with **AC**
        $\mathbf{V}_{out,1} = [\mathbf{F}, \mathbf{A}, \mathbf{E}]$
        $\mathbf{V}_{out,2} = [\mathbf{C}, \mathbf{F}, \mathbf{E}]$
        $\mathbf{V}_{out,3} = [\mathbf{C}, \mathbf{E}, \mathbf{B}]$
        skipTriangles = [**false**, **false**, **false**]
    case 2 or 5 ▷ PLIC intersects **AB** and **BC**
        Calculate **E** ▷ Point intersection of PLIC with **AB**
        Calculate **F** ▷ Point intersection of PLIC with **BC**
        $\mathbf{V}_{out,1} = [\mathbf{E}, \mathbf{B}, \mathbf{F}]$
        $\mathbf{V}_{out,2} = [\mathbf{A}, \mathbf{E}, \mathbf{F}]$
        $\mathbf{V}_{out,3} = [\mathbf{A}, \mathbf{F}, \mathbf{C}]$
        skipTriangles = [**false**, **false**, **false**]
    case 3 or 4 ▷ PLIC intersects **AC** and **BC**
        Calculate **E** ▷ Point intersection of PLIC with **BC**
        Calculate **F** ▷ Point intersection of PLIC with **AC**
        $\mathbf{V}_{out,1} = [\mathbf{E}, \mathbf{C}, \mathbf{F}]$
        $\mathbf{V}_{out,2} = [\mathbf{B}, \mathbf{E}, \mathbf{F}]$
        $\mathbf{V}_{out,3} = [\mathbf{B}, \mathbf{F}, \mathbf{A}]$
        skipTriangles = [**false**, **false**, **false**]
    end
end
return $\mathbf{V}_{out}$, skipTriangles

The cutting operation is recursive and will start with the two initial right triangular faces (see Fig. 51a). To keep the algorithm simple, it is performed in three steps assuming that during each step every triangle is cut during the previous step. The first set of cuts on the fluxing face are made with the intersection defined by $\mathbf{n}^L$ and d^L from the left cell planar interface reconstruction. The cut line can divide each triangle face into three separate triangles and both triangle faces can possibly be cut for a total of six resulting triangles for the second set of cuts (see Fig. 51b). The second set of cuts are due to the intersection of the plane defined by $\mathbf{n}^R$ and d^R from the right cell planar interface reconstruction. After the second set of cuts there may be the possibility to divide each face triangle three more times resulting in up to eighteen final face triangles (see Fig. 51c). During each step, a triangle is cut and the three resulting triangles are stored in the next stage of triangles. If a triangle is not cut, the original triangle is moved to the next stage and the two other non-existent triangles are flagged and skipped during the next stage of cuts. This procedure is outlined in Algorithm 4.



(a) Initial **(b)** After 1st cuts **(c)** After Second Cuts

Figure 51: Cell face flux triangles of volume fraction flux volumes

Algorithm 4 Face Breakdown Algorithm

input $\mathbf{n}^L$, d^L, $\mathbf{n}^R$, d^R

for $i = 1 \rightarrow 2$ **do** ▷ Loop through the initial 2 triangles.
  $\mathbf{V}_{3i:3i+2} = \text{SPLITTRIANGLE}(\mathbf{V}_i, \mathbf{n}^L, d^L)$ ▷ Cut against the 1^{st} PLIC plane.
end for
for $i = 3 \rightarrow 8$ **do** ▷ Loop through the 6 possible *new* triangles.
  if not skipTriangle **then**
    $\mathbf{V}_{3i:3i+2} = \text{SPLITTRIANGLE}(\mathbf{V}_i, \mathbf{n}^R, d^R)$ ▷ Cut against the 2^{nd} PLIC plane.
  end if
end for
for $i = 9 \rightarrow 26$ **do** ▷ Loop through the final 18 possible triangles.
  if not skipTriangle **then**
    fluid = $\text{DETERMINEFLUID}(\mathbf{n}^L, d^L, \mathbf{n}^R, d^R, \mathbf{V})$ ▷ Test triangle for fluid.
    j(fluid) = j(fluid) + 1 ▷ Counter for number of triangles in this fluid.
    $\mathbf{V}_{new}(\text{fluid}, \text{j(fluid)}) = \mathbf{V}_i$ ▷ Sort the triangle into respective fluid.
  end if
end for
return $\mathbf{V}_{new}$

Finally, to determine which of the four flux volume types, $\Omega_{0,0}$, $\Omega_{0,1}$, or $\Omega_{1,0}$, or $\Omega_{1,1}$ each triangle belongs to, the signed distance of the triangle's barycenter to each of the 2 PLIC planes on either side of the cell face is calculated and used to identify the flux volume type, see Algorithm 5.

Algorithm 5 Algorithm to identify triangle $\mathbf{V}$'s flux volume type $\Omega_{0,0}$, $\Omega_{0,1}$, $\Omega_{1,0}$, or $\Omega_{1,1}$

input $\mathbf{n}^L$, d^L, $\mathbf{n}^R$, d^R, $\mathbf{V}$

$\mathcal{P} = \text{BARYCENTER}(\mathbf{V}_{in})$ ▷ Calculate the center point of the triangular face.
$D^L = \text{SIGNEDDISTANCE}(\mathbf{n}^L, d^L, \mathcal{P})$ ▷ Eq. (B.2)
$D^R = \text{SIGNEDDISTANCE}(\mathbf{n}^R, d^R, \mathcal{P})$
type $= 1 + \mathcal{H}(D^L) + 2\mathcal{H}(D^R)$ ▷ $\mathcal{H}$: Heaviside function
return type

The face triangles can now be used to calculate the fluid 1 face area fractions β^L and β^R by simply summing the area of the face triangles belonging to $\Omega_{1,1}$ and $\Omega_{1,0}$ for β^L and $\Omega_{1,1}$ and $\Omega_{0,1}$ for β^R. Furthermore each sorted triangle can now be used to create a flux volume as in Owkes and Desjardins [119], by transporting each triangle vertex along its streak-line back in time using a second-order Runge-Kutta scheme with tri-linear interpolation of face velocities to the vertex location. The face velocities used for this are the contact discontinuity wave speeds λ of the cell face Riemann problem, see Eq. (2.13).

The flux volume thus created is a triangular prism shown in Fig. 52, with a, b, and c the vertices of the face triangle and g, h, and i their transported back in time respective location. To ensure that adjacent flux volumes will have matching side faces and there can be no overlap of simplices, three additional points d, e, and f are introduced as the mid points between the face triangle vertices a, b, and c. These additional points are used for tessellation only, and are not themselves transported. The resulting flux volume tessellation into tetrahedral simplices is given in Table 16.

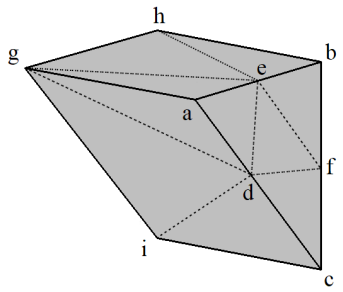


Figure 52: Tessellation of a triangular prism volume fraction flux volume

To ensure conservation and boundedness, a correction volume $\mathcal{V}_{cor}$ for the overall face flux volume is calculated following the procedure of [119]. However, unlike in [119], this correction volume needs to be split among the individual flux volume triangular prisms that form the overall cell face flux volume according to

$$\mathcal{V}_{cor,k} = \frac{(\mathcal{A}_{ghi} \cdot \boldsymbol{e})_k}{\sum_l (\mathcal{A}_{ghi} \cdot \boldsymbol{e})_l} \mathcal{V}_{cor} \tag{B.3}$$

where $\mathcal{V}_{cor,k}$ is the correction volume for the k-th triangular prism flux volume, $(\mathcal{A}_{ghi} \cdot \boldsymbol{e})$ is the projected area of triangle ghi onto the cell face plane, and the sum in the denominator is over all triangular prism flux volumes l that form the cell face flux volume. Once $\mathcal{V}_{cor,k}$ is determined, the location of the vertex o forming the correction tetrahedron $higo$ given in row 8 of Table 16 can be calculated, following a similar procedure as described in [119]. The two coordinates of o not in the normal direction of the cell face are constrained to the barycenter of the triangle ghi, and the coordinate of o in the cell face normal direction is given by enforcing the volume of the correction tetrahedron 8 be equal to the correction volume $\mathcal{V}_{cor,k}$.

Table 16: Simplice Ordering

Simplice	**a**	**b**	**c**	**d**
1	i	d	f	h
2	h	f	e	d
3	i	g	d	h
4	g	d	h	e
5	g	d	e	a
6	h	e	f	b
7	i	f	d	c
8	h	i	g	o

Once each triangular prism flux volume is split into the 8 tetrahedra given in Table 8, it can be fluxed following directly the procedure described in Owkes and Desjardins [119] for individual tetrahedra. However, instead of summing up the contributions from each control volume directly for the entire cell face flux volume as in [119], we need to calculate the contribution from each intersected neighbor control volume to each flux volume type separately, resulting in $\Omega^n_{i,j}$ and $\alpha^n_{i,j}$ with $i = 0, 1$ and $j = 0, 1$ needed in Eqs. (2.31) and (2.33).

To solve Eq. (2.21), the fluid 1 volume fluxing through the entire cell face f in a time step can be calculated from

$$(\Delta\alpha)^f_{cv} = \sum_n \alpha^n_{0,0}\Omega^n_{0,0} + \alpha^n_{0,1}\Omega^n_{0,1} + \alpha^n_{1,0}\Omega^n_{1,0} + \alpha^n_{1,1}\Omega^n_{1,1}\,, \tag{B.4}$$

where the sum is over the 2 by 3 by 3 neighbor cell stencil centered on the cell face. Finally, the right-hand-side of Eq. (2.21) is incorporated with an implicit Euler discretization.

The method outlined above gives results nearly identical to the original method by Owkes and Desjardins [119] in standard interface advection test cases. Small differences are simply due to the additional tetrahedra used to tesselate the overall flux volume and the different way of splitting the correction volume. It should be pointed out that the procedure described above can also be used to geometrically transport the volume-of-fluid scalar on arbitrary polyhedral meshes. The only additional step required is the splitting of the polygonal cell faces into triangles using for example the barycenter of the cell face, and constructing tetrahedral prism flux volumes for each of the triangles thus generated.

APPENDIX C

SPATIAL RECONSTRUCTION SCHEMES

In this section, the various numerical schemes used to reconstruct the left and right primitive states $\boldsymbol{U}_f^L, \boldsymbol{U}_f^R$ for a given face f are described.

C.1 MP-WENO-Z Reconstruction

As described in Section 4.4.3.1 the MP-WENO-Z scheme is a combination of the WENO-Z scheme of Borges et al. [16] and the monotonicity-preserving (MP) bounds of Balsara and Shu [8]. The classical fifth-order accurate WENO reconstruction [148] with a stencil,

$$\begin{aligned} v_f^{\text{WENO-5}} &= \Theta^{\text{WENO-5}}\left(\boldsymbol{U}^3_{\text{stencil},f}\right) \\ &= \Theta^{\text{WENO-5}}\left(\overline{\boldsymbol{U}}_{L-2}, \overline{\boldsymbol{U}}_{L-1}, \overline{\boldsymbol{U}}_{L}, \overline{\boldsymbol{U}}_{R}, \overline{\boldsymbol{U}}_{R+1}, \overline{\boldsymbol{U}}_{R+2}\right) \end{aligned} \tag{C.1}$$

is defined such that,

$$v_f^{K,\text{WENO-5}} = \begin{cases} \omega_0 v_0 + \omega_1 v_1 + \omega_2 v_2 & \text{if } K = L, \\ \tilde{\omega}_0 \tilde{v}_0 + \tilde{\omega}_1 \tilde{v}_1 + \tilde{\omega}_2 \tilde{v}_2 & \text{if } K = R. \end{cases} \tag{C.2}$$

The reconstructed values v_r for the left-biased ($K = L$) reconstruction are obtained from the volume-averages in the rth candidate stencil $\boldsymbol{S}_r = (\overline{v}_{K-r}, \overline{v}_{K-r+1}, \overline{v}_{K-r+2})$,

$$v_0 = \frac{1}{6}\left(-\overline{v}_{R+1} + 5\overline{v}_R + 2\overline{v}_L\right) \tag{C.3a}$$

$$v_1 = \frac{1}{6}\left(2\overline{v}_R + 5\overline{v}_L - \overline{v}_{L-1}\right) \tag{C.3b}$$

$$v_2 = \frac{1}{6}\left(11\overline{v}_L - 7\overline{v}_{L-1} + 2\overline{v}_{L-2}\right) \tag{C.3c}$$

The nonlinear weights ω_r read,

$$\omega_r = \frac{\alpha_r}{\sum_r \alpha_r}, \qquad \alpha_r = \frac{\gamma_r}{(\epsilon + \beta_r)^2} \tag{C.4}$$

The small parameter $\epsilon = 1 \times 10^{-6}$ [73] is used to keep the non-linear weights bounded and to prevent division by zero. The linear weights are given by,

$$\gamma_0 = \frac{3}{10}, \qquad \gamma_1 = \frac{3}{5}, \qquad \gamma_2 = \frac{1}{10} \tag{C.5}$$

The nonlinear weights w_r are derived from the ideal weights γ_r by taking into account the stencil smoothness. The latter is characterized by the smoothness indicator β_r of [73],

$$\beta_0 = \frac{13}{12}\left(\overline{v}_L - 2\overline{v}_R + \overline{v}_{R+1}\right)^2 + \frac{1}{4}\left(3\overline{v}_L - 4\overline{v}_R + \overline{v}_{R+1}\right)^2 \tag{C.6a}$$

$$\beta_1 = \frac{13}{12}\left(\overline{v}_{L-1} - 2\overline{v}_L + \overline{v}_R\right)^2 + \frac{1}{4}\left(\overline{v}_{L-1} - \overline{v}_R\right)^2 \tag{C.6b}$$

$$\beta_2 = \frac{13}{12}\left(\overline{v}_{L-2} - 2\overline{v}_{L-1} + \overline{v}_L\right)^2 + \frac{1}{4}\left(\overline{v}_{L-2} - 4\overline{v}_{L-1} + 3\overline{v}_L\right)^2 \tag{C.6c}$$

The right-biased ($K = R$) reconstruction follows in analogous fashion by exploiting the symmetry with respect to the face location.

The WENO-Z reconstruction [16] refers to a novel family of WENO schemes, which aim to introduce lower dissipation and provide higher accuracy than the classical WENO schemes [73]. An improved smoothness indicator referred to as the global smoothness indicator is introduced [16],

$$\tau_5 = |\beta_0 - \beta_2| \tag{C.7}$$

The nonlinear weights read,

$$\omega_r = \frac{\tilde{\alpha}_r}{\sum_r \tilde{\alpha}_r}, \qquad \tilde{\alpha}_r = \gamma_r\left(1 + \frac{\tau_5}{\beta_r + \epsilon}\right) \tag{C.8}$$

with $\epsilon = 10^{-40}$. The monotonicity-preserving bounds for the reconstruction following [8] is given by,

$$v_f^{L,\mathrm{MP}} = v_f^{L,\text{WENO-Z}} + \mathrm{minmod}\left(v_f^{L,\min} - v_f^{L,\text{WENO-Z}}, v_f^{L,\max} - v_f^{L,\text{WENO-Z}}\right) \tag{C.9}$$

where the generalized minmod function is

$$\mathrm{minmod}\left(a_1, a_2, \ldots, a_n\right) = \begin{cases} \min\left(a_1, a_2, \ldots, a_n\right) & \text{if } \forall_i : a_i > 0, \\ \max\left(a_1, a_2, \ldots, a_n\right) & \text{if } \forall_i : a_i < 0, \\ 0 & \text{otherwise.} \end{cases} \tag{C.10}$$

The minimum and maximum bounds denoted by $v_f^{L,\min}$ and $v_f^{L,\max}$ are given by,

$$v_f^{L,\min} = \max\left(\min\left(\overline{v}_L, \overline{v}_R, v_f^{\mathrm{MD}}\right), \min\left(\overline{v}_L, v_f^{\mathrm{UL}}, v_f^{\mathrm{LC}}\right)\right) \tag{C.11a}$$

$$v_f^{L,\max} = \min\left(\max\left(\overline{v}_L, \overline{v}_R, v_f^{\mathrm{MD}}\right), \max\left(\overline{v}_L, v_f^{\mathrm{UL}}, v_f^{\mathrm{LC}}\right)\right) \tag{C.11b}$$

where the upper limit (UL), median (MD) and large curvature (LC) bounds are,

$$v_f^{\mathrm{UL}} = \overline{v}_L + 2\left(\overline{v}_L - \overline{v}_{L-1}\right) \tag{C.12a}$$

$$v_f^{\mathrm{MD}} = \frac{1}{2}\left(\overline{v}_L + \overline{v}_R\right) - \frac{1}{2}\tilde{d}_f \tag{C.12b}$$

$$v_f^{\mathrm{LC}} = \overline{v}_L + \frac{1}{2}\left(\overline{v}_L - \overline{v}_{L-1}\right) + \frac{4}{3}\tilde{d}_f \tag{C.12c}$$

The curvature is measured by the undivided difference operator,

$$d_L = \overline{v}_R - 2\overline{v}_L + \overline{v}_{L-1} \tag{C.13}$$

Setting

$$\tilde{d}_f = \mathrm{minmod}\left(4d_L - d_R, 4d_R - d_L, d_L, d_R, d_{L-1}, d_{R+1}\right) \tag{C.14}$$

filters out extremal features that have very small domain of support, but leaves extremal features with large support intact.

C.2 TVD-MUSCL Reconstruction

The second-order accurate TVD-MUSCL reconstruction [88, 163] with a stencil,

$$\begin{aligned} v_f^{\text{TVD-MUSCL}} &= \Theta^{\text{TVD-MUSCL}}\left(\boldsymbol{U}^2_{\text{stencil},f}\right) \\ &= \Theta^{\text{TVD-MUSCL}}\left(\overline{\boldsymbol{U}}_{L-1}, \overline{\boldsymbol{U}}_L, \overline{\boldsymbol{U}}_R, \overline{\boldsymbol{U}}_{R+1}\right) \end{aligned} \tag{C.15}$$

is defined such that,

$$v_f^{K,\text{TVD-MUSCL}} = \begin{cases} \overline{v}_L + \frac{1}{2}\sigma_L & \text{if } K = L, \\ \overline{v}_R - \frac{1}{2}\sigma_R & \text{if } K = R. \end{cases} \tag{C.16}$$

where σ_L, σ_R are the limited slopes. Following Titarev and Toro [163], a minmod-type limiter [82] is utilized, where,

$$\sigma_{cv} = \frac{1}{2}\Big(\mathrm{sign}\left(\Delta^- \overline{v}_{cv}\right) - \mathrm{sign}\left(\Delta^+ \overline{v}_{cv}\right)\Big)\,\mathrm{minmod}\Big(\left|\Delta^- \overline{v}_{cv}\right|, \left|\Delta^+ \overline{v}_{cv}\right|\Big) \tag{C.17}$$

The left- and right-biased undivided difference operators are given by,

$$\Delta^- \overline{v}_i = \overline{v}_i - \overline{v}_{i-1} \tag{C.18a}$$

$$\Delta^+ \overline{v}_i = \overline{v}_{i+1} - \overline{v}_i \tag{C.18b}$$

C.3 THINC Reconstruction

In one-dimensional space x, the volume fraction for a mesh cell i is defined such that

$$\alpha_i(t) = \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_{i+1/2}} \xi(x,t)\, \mathrm{d}x \tag{C.19}$$

where $x_{i-1/2}, x_{i+1/2}$ refer to the faces in the negative and positive x directions, Δx is the local mesh spacing and ξ denotes the indicator function for a well-resolved, sharp phase interface,

$$\xi(x,t) = \begin{cases} 1 & \text{if } x \in \Omega_1 \\ 0 & \text{if } x \in \Omega_2 \end{cases} \tag{C.20}$$

where Ω_1, Ω_2 are the volumes occupied by the two fluids under consideration. The THINC scheme aims to replace the Heaviside-type characteristic function Eq. (C.20) by the locally defined sigmoid type function within the definition of the volume fraction Eq. (C.19), such that

$$\xi_i(x,t) = \frac{1}{2}\left[1 + \tanh\left(\beta_i \left(\delta_i \frac{x - x_{i-1/2}}{\Delta x} + \tilde{x}_i\right)\right)\right] \quad \in [0,1] \tag{C.21}$$

where $\delta_i = \mathrm{sign}\,(\alpha_{i+1} - \alpha_{i-1})$ is the sign of the volume fraction gradient, the parameter β_i controls the interface thickness and $\tilde{x}_i$ denotes the interface offset with respect to the current mesh cell. The interface offset $\tilde{x}_i$ is uniquely determined from the volume fraction Eq. (C.19) and the definition of the hyperbolic tangent function Eq. (C.21) as follows,

$$\tilde{x}_i = \frac{1}{2\beta_i} \ln\left(\frac{B-1}{A-B}\right) \tag{C.22}$$

with $A = \exp(2\delta_i\beta_i)$ and $B = \exp(2\delta_i\beta_i\alpha_i)$ [45]. The reconstruction of the volume fraction at the cell faces are then given by,

$$\alpha_{i+1/2}^{K,\mathrm{THINC}} = \begin{cases} \frac{1}{2}\left(1 + \tanh\left(\beta_i\left(\delta_i + \tilde{x}_i\right)\right)\right) & \text{if } K = L, \\ \frac{1}{2}\left(1 + \tanh\left(\beta_{i+1}\tilde{x}_{i+1}\right)\right) & \text{if } K = R \end{cases} \tag{C.23}$$

With the volume fraction (i.e. α_1) reconstructed at a cell face using Eq. (C.23), the remaining variables (i.e. $\rho_1\alpha_1, \rho_2\alpha_2$) which depend on the variation of the volume fraction must also be treated consistently. Shyue and Xiao [155] followed So, Hu, and

Adams [158] and devised a homogeneous-equilibrium-consistent reconstruction scheme for the remaining conservative variables $\boldsymbol{Q}$, see Eq. (4.15). As noted by [164], treating the remaining variables in conservative form could lead to spurious oscillations near the material interface. To circumvent this, [45] proposed an integral formulation of the phasic densities (i.e. $\rho_1\alpha_1, \rho_2\alpha_2$) that is consistent with the reconstruction of the volume fraction. In the present work, the approach of Shyue and Xiao [155] is followed for the primitive variables $\boldsymbol{U}$ rather than the conservative variables $\boldsymbol{Q}$. A consistent treatment of the phasic densities reads,

$$(\alpha_l\rho_l)^{K,\mathrm{THINC}}_{i+1/2} = \begin{cases} (\alpha_l\rho_l)_i \quad + (\rho_l)_i \left((\alpha_l)^{K,\mathrm{THINC}}_{i+1/2} - (\alpha_l)_i \right) & \text{if } K = L, \\ (\alpha_l\rho_l)_{i+1} + (\rho_l)_{i+1} \left((\alpha_l)^{K,\mathrm{THINC}}_{i+1/2} - (\alpha_l)_{i+1} \right) & \text{if } K = R \end{cases} \tag{C.24}$$

for $l = 1, 2$. In the absence of surface tension, there is no dependency of velocity and pressure on the volume fraction and hence, these quantities can be reconstructed using the polynomial reconstruction operator Θ^{PR}. However, surface tension introduces an implicit dependency of pressure on the volume fraction due to the presence of the gradient of the volume fraction. This mandates that the pressure be also treated consistently with the volume fraction as,

$$p^{K,\mathrm{THINC}}_{i+1/2} = \begin{cases} p_i \quad + (\sigma\kappa)_i \left((\alpha_1)^{K,\mathrm{THINC}}_{i+1/2} - (\alpha_1)_i \right) & \text{if } K = L, \\ p_{i+1} + (\sigma\kappa)_{i+1} \left((\alpha_1)^{K,\mathrm{THINC}}_{i+1/2} - (\alpha_1)_{i+1} \right) & \text{if } K = R \end{cases} \tag{C.25}$$

The genuinely one-dimensional hyperbolic tangent function described so far is applied to multi-dimensional problems by scaling the interface thickness parameter in accordance to the orientation of the material interface [155, 45],

$$\beta_i = \beta \left| n^x_i \right| + 0.01 \tag{C.26}$$

for the x direction and analogously for the other directions. Here n^x denotes the $x-$component of the normal vector $\boldsymbol{n} = (n^x, n^y, n^z)^T$, the calculation of which is described in Section 4.4.4. Throughout this work, a value of $\beta = 3$ is used, which limits the interface thickness between two and three cells based on the chosen $\epsilon_{\mathrm{THINC}}$ value, Eq. (4.24). The reader is directed to the work of Xiao, Ii, and Chen [175] for discussion on how the selection of β and $\epsilon_{\mathrm{THINC}}$ relates to the thickness of the diffused region of the phase interface.

APPENDIX D

APPROXIMATE HLLC RIEMANN SOLVER

As described in Section 4.4.3.1, the convective numerical flux $\boldsymbol{F}_{\text{conv},f}$ requires the solution of a Riemann problem Eq. (4.19) at every cell face. In this work, a Harten-Lax-van Leer-Contact (HLLC) type approximate Riemann solver [165] is used to solve Eq. (4.19). Specifically, the HLLC solver initially used in the context of the five-equation model by [28] and later extended by [46] to include capillary effects [123] is utilized here.

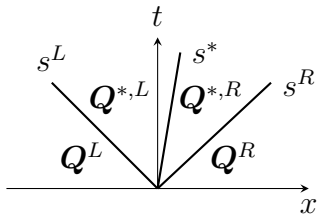


Figure 53: Riemann fan for a typical three-wave HLLC approximate Riemann solver, with the left and right vector of conservative variables $\boldsymbol{Q}^L$ and $\boldsymbol{Q}^R$, and the left and right acoustic wave speeds s^L and s^R. The left and right intermediate states $\boldsymbol{Q}^{*,L}$ and $\boldsymbol{Q}^{*,R}$ are separated by the intermediate wave velocity s^*.

The numerical flux at a cell face f using the left and right vector of conservative variables $\boldsymbol{Q}_f^L, \boldsymbol{Q}_f^R$ reads in a compact notation [74, 28],

$$\begin{aligned}\widehat{\boldsymbol{F}}_{\text{conv},f} = &\frac{1+\text{sign}\left(s^*\right)}{2}\Big(\boldsymbol{F}^L + s^-\left(\boldsymbol{Q}^{*,L} - \boldsymbol{Q}^L\right)\Big) \\ &+ \frac{1-\text{sign}\left(s^*\right)}{2}\Big(\boldsymbol{F}^R + s^+\left(\boldsymbol{Q}^{*,R} - \boldsymbol{Q}^R\right)\Big)\end{aligned} \tag{D.1}$$

where $s^{\pm}$ are the acoustic wave speeds and s^* is the estimated wave speed in the intermediate region, see Fig. 53. The states in the intermediate region $\boldsymbol{Q}^{*,K}$ for $K = L, R$ are given by,

$$\boldsymbol{Q}^{*,K} = \left(\frac{s^K - u^K}{s^K - s^*}\right) \begin{pmatrix} (\rho_1\alpha_1)^K \\ (\rho_2\alpha_2)^K \\ \rho^K s^* \\ (\rho v)^K \\ (\rho w)^K \\ (\rho E)^K + \left(s^* - u^K\right)\left(\rho^K s^* + \frac{p^K - \sigma\bar{\kappa}(\alpha_1)^K}{s^K - u^K}\right) \\ (\alpha_1)^K \end{pmatrix} \tag{D.2}$$

where $\bar{\kappa}$ denotes the arithmetic average of the interface curvature. The introduction of surface tension effects in the HLLC Riemann solver follows the work of Garrick, Owkes, and Regele [46] wherein the pressure jump across the intermediate state is taken as,

$$p^{*,R} - p^{*,L} = \sigma\kappa\left(\alpha_1^R - \alpha_1^L\right) \tag{D.3}$$

The waves speeds used in Eqs. (D.1) and (D.2) are obtained following [35, 9],

$$s^- = \min\left(0, s^L\right), \qquad s^L = \min\left(\bar{u} - \bar{c},\ u^L - c^L\right) \tag{D.4a}$$

$$s^+ = \max\left(0, s^R\right), \qquad s^R = \max\left(\bar{u} + \bar{c},\ u^R + c^R\right) \tag{D.4b}$$

where $\bar{u}$ and $\bar{c}$ are the Roe- or arithmetic-averages of the left and right located velocity and speed of sound respectively. In this work, the less computationally demanding arithmetic averages are used, similar to [28, 46, 45]. The estimate of the wave speed in the intermediate region is given by [46],

$$s^* = \frac{p^R - p^L + \rho^L u^L\left(s^L - u^L\right) - \rho^R u^R\left(s^R - u^R\right) - \sigma\bar{\kappa}\left(\alpha_1^R - \alpha_1^L\right)}{\rho^L\left(s^L - u^L\right) - \rho^R\left(s^R - u^R\right)} \tag{D.5}$$

Finally, the HLLC solver can be used to upwind certain quantities of interest to be used in the discretization of the viscous and source terms proposed originally by Coralic and Colonius [28] and then used later in [46, 45]. The velocity components of

the left and right states upwinded to the cell face using,

$$u_f = \frac{1+\text{sign}\left(s^*\right)}{2}\left(u^L + s^-\left(\frac{s^L-u^L}{s^L-s^*}-1\right)\right) + \frac{1-\text{sign}\left(s^*\right)}{2}\left(u^R + s^+\left(\frac{s^R-u^R}{s^R-s^*}-1\right)\right) \tag{D.6a}$$

$$v_f = \frac{1+\text{sign}\left(s^*\right)}{2}\,v^L + \frac{1-\text{sign}\left(s^*\right)}{2}\,v^R \tag{D.6b}$$

$$w_f = \frac{1+\text{sign}\left(s^*\right)}{2}\,w^L + \frac{1-\text{sign}\left(s^*\right)}{2}\,w^R \tag{D.6c}$$

and the volume fraction upwinded to the face is given by,

$$\alpha_{1,f} = \frac{1+\text{sign}\left(s^*\right)}{2}\,\alpha_1^L + \frac{1-\text{sign}\left(s^*\right)}{2}\,\alpha_1^R \tag{D.7}$$