5-11-2023

A Novel Analytic Linear Interface Calculation and Its Use in Computational Fluid Dynamics for Enhanced Multiphase Modeling

Simon P. Shipkowski
sps3287@rit.edu

RIT

A Novel Analytic Linear Interface Calculation and Its Use in Computational Fluid Dynamics for Enhanced Multiphase Modeling

by

Simon Peter Shipkowski

A dissertation submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Microsystems Engineering

Microsystems Engineering Program
Kate Gleason College of Engineering

Rochester Institute of Technology
Rochester, New York
May 11, 2023

A Novel Analytic Linear Interface Calculation and Its Use in Computational Fluid Dynamics for Enhanced Multiphase Modeling
by
Simon Peter Shipkowski

Committee Approval:

We, the undersigned committee members, certify that we have advised and/or supervised the candidate on the work described in this dissertation. We further certify that we have reviewed the dissertation manuscript and approve it in partial fulfillment of the requirements of the degree of Doctor of Philosophy in Microsystems Engineering.

Dr. Isaac Perez-Raya — Date
Assistant Professor, Mechanical Engineering (Advisor)

Dr. Kara Maki — Date
Associate Professor, Mathematical Sciences

Dr. Michael Schertzer — Date
Associate Professor, Mechanical Engineering

Dr. Patricia Taboada-Serrano — Date
Associate Professor, Chemical Engineering

Certified by:

Dr. Stefan Preble — Date
Director, Microsystems Engineering Department

Abstract
Kate Gleason College of Engineering
Rochester Institute of Technology

Degree: Doctor of Philosophy **Program:** Microsystems Engineering
Authors Name: Simon Peter Shipkowski
Advisors Name: Dr. Isaac Perez-Raya
Dissertation Title: A Novel Analytic Linear Interface Calculation and Its Use in Computational Fluid Dynamics for Enhanced Multiphase Modeling

Interfacial modeling is of widespread interest in manufacturing, imaging, and scientific applications. Presented here is an interfacial reconstruction method designed for use in Computational Fluid Dynamics (CFD) simulations of nucleate boiling, yet it is more broadly applicable. Nucleate boiling is vital in current and emerging technologies due to the high heat transfer at low temperature.

A challenge for multiphase CFD simulation with phase change mass flux is the competing interests of quality mass transfer conservation, resolution of interfaces, and computational time/power. This Piecewise Linear Interface Calculation-Analytic Size Based (PLIC-ASB) method was developed to mitigate this problem in the volume of fluid simulation method so that all three ends could be achieved simultaneously.

Research consisted of a cyclical process of testing in an expanding parameter space to ensure improvement to the PLIC-ASB, the CFD simulation, or the testing itself. This parameter space includes data from: theoretical, numerical, and empirical sources; interface methods, and options (e.g. gradient methods); integration routine sensitivity analysis, and alternate software instantiation.

The main findings are: (1) the PLIC-ASB yields high levels of accuracy that converge to theoretical and empirical results as the mesh size is refined. The relative error reduction tends to be over 70% compared to the relevant industry standard. (2) The PLIC-ASB exceeded the design parameters. Symbolic formulation converts easily to code and its exactness permits a sharp interface. (3) The PLIC-ASB was less prone to error due to gradient method changes.

Acknowledgments

My thanks for the gift of nature and to her Author.

My deep thanks to:
My advisor Professor Perez-Raya, who gave a lecture in class on his research and ever since we have been working together. I thought every other student would be rushing to collaborate with you because I could see what they missed. When I developed this method I did not think it was important but you encouraged me because you could see what I missed. It has been an honor to collaborate with you. I am very grateful.

The professors of the committee
Professor Taboada-Serrano who was part of all of my major degree parts, your input has helped me to continue to remember why the research is of interest to me. Professor Schertzer, who has had the right question even when it was not easy. Professor Maki, who helped me gain confidence due to revisions that were my first draft, but mostly kept me humble with the rest of them. Professor Cromer, for your time and care in the exam.

RIT and the Microsystems Engineering Department, Faculty and Staff for the scholarship, the continued support, and an engaging community especially, Director Preble, for your time since my initial interview. Your professionalism and dedication to sharing academically is a gift to us all. Professor Smith, for my interview and the time to speak with you about this great program. Lisa, who is eager to help so many yet we all feel as if we are your personal friend due to your kindness. Dr. Lewis, for the arrangements at the dissertation exam.

My former advisor and lab group
Professor Kandlikar, Daino, Ankit, Alyssa, Aniket, Carlos, Maharshi and many more! It was an honor to be part of this lab and to begin research with Professor Perez-Raya because of it.
T he LL-lab of Mert, Mario, and Yu Kee as well as all other RIT colleagues.
You know what to do with your cheese cakes deliveries.

These key figures: Dr. Clidas ESP, for being an external mentor professor and ESP. Skelly as a mentor, (Dr. Morris, D.Zuber, Dr. Mack, and Terri) whose work was integral to my progress.

Friends and Family
My parents, Arthur, Steph^{d}, Jim and Carol, Aaron, Sam, Corinne, John, Tyler, Karla, Miguel, Kate, Logan, Chris, Glen, and RACWI Members

Prior educators and colleagues
Professor Iashvili, Tanveer, Joe, Andrew, Joseph, James, Mark, Adam, Jimmy, Tai Lung, Ben, Ian, Jason, Steve, Adam and Jonathan Professors Tahar, Fazar, Boersma, Martin, Gamory, and Waddach Stefan, Tim, Steve and Steve and Steve

My most profound thanks to:
My most beautiful and caring wife for having assisted me so greatly with all of my research and after the injury. Thank you, Jennifer Shipkowski

If I have seen further. . .

In memory of my personal giants and loved ones.

London, Cindy, Andy, Gran, Pop, Helen, Janet,
Lu, Higbie, Dave, Professor Baur, Doc (Professor Mancuso),
and Professor Jan Wirnowski

Contents

List of Figures

List of Tables

Nomenclature

A	Hamaker constant (J)
A_{int}	Interface surface area (m^2)
c	Specific heat (J/kg-K)
$coeff$	Empirical Coefficient
d	Distance interface to cell center (m)
dx	Length of cell side (m)
F	Volume fraction
f	Evaporation coefficient
g	Gravitational constant (m/s^2)
h_{fg}	Latent heat of evaporation(J/kg)
K	Curvature (1/m)
k	Thermal conductivity (W/m-K)
k_l	Thermal conductivity of the liquid (W/m K)
l_{int}	Length of interface (m)
M	Molecular weight (kg/mol)
m''_{mic}	Interfacial mass flux in the microlayer

m''	Mass flux (kg/s-m^2)
$\hat{n}$	Unit normal vector
n_x	Normal vector component
n_y	Normal vector component
P_l	Liquid pressure in microlayer (Pa)
P_v	Pressure vapor (Pa)
p	Pressure (Pa)
$\dot{Q}$	Integrated heat flux (W/m)
q''	Heat flux (W/m^2)
R	Universal gas constant (J/mol-K)
R_b	Bubble radius (m)
R_{int}	Interfacial resistance (m^2-K/W)
r	Radial coordinate (m)
R	Arbitrary Orthogonal rotation matrix
$S_t(t_x, t_y)$	Smaller angle selecting step function
S_M	Momentum source term (N/m^3)
S_T	Energy source term (K/s)
S_{mic}	Mass source term evaporation (kg/s-m^3)
s_d	Distance interface to cell center (m)
T	Temperature (K)
T_{sat}	Saturation temperature(K)
$T_{w,mic}$	Surface temperature at the microlayer (K)

$\overrightarrow{t}$	Interface-tangential vector
t	Time (K)
t_x	Interface-tangential vector, x-component
t_y	Interface-tangential vector, y-component
u	Cartesian x or radial velocity (m/s)
V	Volume (m/s)
$V_{m,l}$	Liquid molar volume (m^3/mol)
v	Fluid velocity (m/s)
v_{int}	Fluid velocity at the interface (m/s)
x	Cartesian x coordinate (m)
y	Cartesian y coordinate (m)
z	Axial coordinate (m)
α	Thermal diffusivity (m^2/s)
α_1	Volume fraction, phase-1
α_2	Volume fraction, phase-2
$\alpha_{,inv}$	Volume fraction, invariant form
β	Interface displacement constant (m^2/s)
δ	Microlayer thickness (m)
ϵ	Density ratio
η	Microlayer axial coordinate (m)
Γ	Mass flux per unit contact line length (kg/s-m)
μ	Dynamic viscosity (Pa-s)

ν	Kinematic viscosity (m^2/s)
ϕ	distance from LS cell to interface (m)
ρ	Density (kg/m^3)
ρ_l	Liquid density (kg/m^3)
ρ_v	Vapor density (kg/m^3)
σ	Surface tension (N/m)
θ	Angle from vertical boundary (rad)
φ	Apparent contact angle (rad)
ξ	Microlayer radial coordinate (m)
ad	Adsorbed film
c	Cell
cl	Contact line
l	Liquid
lv	Liquid-vapor interface
mic	Microlayer
sat	Saturation
v	Vapor
w	Wall

Chapter 1

Introduction

Quantification, measurement, and reconstruction processes through computational means is the primary direction of this research. These processes have been conducted in computational fluid dynamics and related data processing with interest in the boiling phenomenon. The improvement of numerical simulations thereof by means of advancing and testing simple user defined functions has been the focus. By simple, the exclusion of higher-order fits and approximation methods is intended and the use of symbolically expressed functions with analytic solutions is also intended.

Utilizing computer modeling to explain experimental observations enables a pathway to further increasing the performance of multiphase flow applications [23]. Computational methods must represent both anticipated phenomenology per available data and be consistent with represent-

ing these outcomes via a proper mechanistic application. This is to say if no application of the mechanistic relations is used to represent available data a simulation is maybe just an animation. At the same time proper application of the physical theories which results in a disconnect from data without an understanding of the source of these difference is also not very useful.

Computational methods have been proposed to allow computer modeling to capture fundamental transport mechanisms in multiphase flows. Successful methods employ traditional discretization techniques, analytical mathematics, and numerical methods that can be implemented by the community at large. The present work improves computer modeling of multiphase flows by proposing a Piecewise linear interface calculation – analytical size-based (PLIC-ASB) method to accurately calculate interface surface area within a computational cell; this in turn, enhances mass transfer calculations by resolving the related local interfacial attributes.

Piecewise Linear Interface Calculation (PLIC) is the general type of any calculation that uses a linear interface to represents the interface in a computational cell. This general form of interface calculation is piecewise if no continuity condition is imposed at the cell to cell boundary.

Multiphase simulation is a challenging field due to the many degrees of freedom involved. The interface of two phases is a mechanically complicated arrangement. Adding in heat and phase changing mass transfer, and the matter is even further complicated. Due to the complexity there is a degree of reliance on the existing platforms and the need to customize to the particulars of a research interest. The use of partition function type mechanics in developing a system of individual particles (as some simulations do) can be misleading in that the behavior of the

interface shape at macroscopic sizes is of interest in much of CFD and not in those simulation. As some prior results are included, some portions of the author's prior related publications have been adapted to the introduction and background so as not neglect any salient detail.

In the CFD multiphase research community it is the need to a great number of calculations which are kept in balance as they are approximations or estimations. It can be too easily left off here with an understanding that it is a matter of cpu and real world time that is all that limits the highest quality simulations. In some sense that is true, yet there the purpose of advancing theories and algorithms is not simply stretching the envelop a few percent. There is a mechanical metaphor commonly used to explain the computational solving algorithm, a damped spring. An under damped spring could be like a high powered computer without a good algorithm, although it quickly approaches the solution it is not well controlled. An over-damped spring is like an excessive algorithm, it has no chance of missing the solution at the expense of time. The perfectly damped system balances resources to elegantly solve as quickly as possible. This metephor is nice, but as it only points to how this complex problem is where balance and nuance are needed for advancement.

Currently, computer simulations employ effective methods to realistically model multiphase phenomena. Osher and Sethian [77] proposed the Level-set method (LS) interface tracking method, and Hirt and Nichols [41] proposed the Volume-of-Fluid (VOF) interface tracking method. The LS and the VOF methods are widely employed techniques to track interfaces in multiphase modeling. Brackbill et al. [4] proposed a continuous surface force (CSF) model that

is a pillar in modeling fluid behavior in multiphase flows. Gibou et al. [33, 34] described the ghost fluid method that effectively includes the interface saturation temperature by fixing the temperature of the interface-cells based on the interface location and temperature. Wayner et al. [127, 128, 129] developed a theoretical model to determine the local heat transfer coefficient in a liquid film that often appears in boiling simulations to account for microlayer effects. These methods are shown useful and viable when they are realistic (explain experimental findings and extend to multiple use cases) and suitable for community application.

Utilizing mechanistic-based modeling continues to advance understanding of experimental results and increases predictive capacity. Proposed methods have contributed to identifying microscale characteristics and to accurately predict the overall performance of heat transfer systems [23, 56]. Dhir et al. [7, 114, 115, 133] utilized LS interface tracking and proposed a microlayer modeling approach to simulate boiling and identified the bubble growth and heat transfer mechanisms at various operating conditions, including high heat fluxes and microgravity. Stephan et al. [40, 60, 61, 88] simulated multiphase phenomena (nucleate boiling and droplet impingement) and proposed a microlayer model to identify the heat transfer in the solid heated plate. Bardia and Trujillo [10] modeled the effect of pressure on the saturation temperature and identified the contribution of inertia forces on bubble growth and heat transfer. Sato and Ničeno [97, 98, 99] performed supercomputer modeling to predict the pool-boiling curve for a heat sink with micropillars. Yazdani et al. [136] performed advanced 3D simulations of flow boiling in microchannels and numerically estimated the boiling curve. Although progress has

been made in the predictive power and overall performance of boiling and heat transfer simulations through the accurate representation of multiphase phenomena at the interface level, there remains open space for new methods that will improve this correlation with experimental findings due to causality. These methods will enable simulations to explain observed enhancement mechanisms [55].

Methods used for interface area calculation tend to be a tradeoff of performance, computational expense, and implementation complexity. The empirical coefficients method [15, 62, 138] is notable for being informed by experimental results. As such, it can be tuned to coincide with data however, it lacks the capacity for resolving fine details of the interface at any particular point. Beside a reasonable bulk estimate of an otherwise known setup it is lacking. The VOF gradient method [28, 59, 102] can resolve sharp interfaces and more details, however it remains an estimate. It is sensitive to the gradient method used for its calculation and is prone to a degree of systemic error requiring a skilled researcher to overcome. This error is due to the method being an approximation based on calculus results. This means that a gradient method which only uses one-cell-step difference will yield good results usually, however tearing and interface artifacts will present without additional preventative measures such as use of the segregation method (see methods section). The interface reconstruction method [14, 35, 141] offers the best outcomes however there is not enough details in the literature to recreate these results and of the details given it is complicated.

Other methods used to simulate interfaces in CFD where multiphase mass transfer is an integral

component, include: nodal or marker [42, 71, 100], mesh refining or mesh-gradient [79], and interface-centric techniques such as the level-set technique [8, 30, 89]. The nodal and mesh refining techniques are roughly just methods to use more computational power in the previously mentioned tradeoff balance as they refine the mesh (increase power/time) for better results albeit in a systematic way. In the specific case of Youngs' algorithm [71], reconstructing thin strands below the cell size as a notable feature for application specific use. Interface-centric techniques may have outstanding results for interface quantification with the challenges of user complexity, imprecise mass transfer, and entire simulation methodology differing than that being presented here. More will be detailed in the Background section, but the challenge to the user is in setting a simulation or problem geometry for these methods (at least generally). The imprecise mass conservation comes in from re-initialization methods or geometric limits, and the simulation methodology refers to the classification of Lagrangian versus Eularian methods. Coupled methods of Level-set and VOF have been offered as well [16, 113, 122] with promising results and relying on researcher implementation.

Although there are all of the above methods and many others, an interface calculation which can yield a sharp interface, does not disrupt strict mass conservation, all while not requiring out of the ordinary computational power and or an intensive user implementation process had not presented, at least not with any fitting notoriety. Presented here is the method that meets these requirements. The PLIC-ASB method, which is not only able to meet all of the aforementioned challenges, thus making it an ideal option for a default PLIC method in standard use for mul-

tiphase VOF simulation schemes with mass flux, but this method is a computationally simple algorithm as well. Without any reliance on iterations or approximations the equivelent linear interface size within a given cell can be determined as a function of the volume fraction of the cell in question and that cell's normal vector (or, the volume fractions of the surrounding cells) as the only argument in the computational function. These two latter parts are essentially equivalent through the application of the gradient. As the solution is analytical, it is theoretically validated. Further evidence of its validity will be given by its results in this document. This is not the end of the story however in the software validation and verification process. Further verification of the algorithm was performed on a hand selected samples and a few test cases in ANSYS-Fluent, such as circular static or growing bubbles and other more advanced multiphase geometries. The results indicate an order of magnitude of improvement in most cases of relative error from VOF gradient methods that use identical arguments in their computational functions. Taking identical arguments without adding computational complexity or expense makes the implementation of this method on a reasonable choice. These data are presented as a running narrative of the document, with results on pertinent cases. In contrast, other corrections to the problem have had expensive interpolative methods and complex programming barriers to their implementation. These higher order corrections may be easy for some to understand, but they can also be a complete impediment or at least a wonder-wall challenge for inexperienced early researcher. The PLIC-ASB method presented here is useful beyond simulations in the multiphase heat transfer area of research. However boiling related interests are emphasized with specific notes on how the given item relates to or motivates simulation-experimental set

ups, as boiling research is the most pertinent to the continued contribution of this research and our motivation for the foundational publication on the subject for other researchers.

1.1. Hypothesis and Objectives

The goal of this body of research is to advance through examination and testing, a novel method for calculating interfacial area in the Volume of Fluid (VOF) computational Fluid Dynamics (CFD) scheme. This research should address the lack of such a method.

Design Standards

The method will be size of accuracy, implementation-simple, computationally reasonable, and a mass conserving.

That also will have a comprehensive implementation description.

For details on the design of the method please see section 3.1.3.

1.1.1. Hypothesis

The Hypothesis:

These following items have a direct 1:1 connection to the testable objectives is used to confirm or deny the hypothesis.

1. The design standards will be met

   Method must be size of accuracy, implementation-simple, computationally reasonable, and a mass conserving.

2. Functionality will be verified

   Computational verification to test its argument data (input) for program errors and that the method is calculating as reported mathematically (output)

3. The validity will be shown

   The developed method will improve interface size accuracy averaging to at least a one-half reduction in amount of error across comparative testing by the available methods used in its place. Specificlly the VOF gradient method as interface reconstrution methods are not available in the literature. This represents a comparison with the state of the art mass conserving technique. Simulations compared with theoretical values.

4. Evidence for consistent improved accuracy will be found.

   The developed method will agree with theoretical and empirical data, and reduce the VOF gradient error rate significantly (by one-half).

1.1.2. Objectives

The Objectives of this Body of Research:

1. **Develop a VOF interface calculation method that meets these Standards:**

   More accurate than comparable alternatives (see below)

   Able to produce a sharp deformation free interface

   Easy to implement

   Insensitive to discrete gradient (normal vector calculation)

   Computationally fast

2. **Test the developed calculation method to Verify functionality on:**

   Simple case inputs (user error detection) Here, after initial by hand data testing is done, large scale data is used where no errors should be found to detect if there are strange function input data being generation rarely.

   All possible inputs (method error detection) Here, after being more confident in the form of the input the method is tested on all forms of data and including the shapes that have extreme features inorder to find if bad output can be found that is unanticipated.

3. **Test developed calculation method to Validate on:**

   Shapes of known sizes

   Time dependent simulations(adiabatic)

4. **Use the developed calculation method in trial of Accuracy against comparable VOF gradient method:**

   Static shapes

   Time dependent simulations (adiabatic)

Introduction

Time dependent simulations with empirical confirmation (non-adiabatic)

Simulations of Increased complexity (heated wall)

Chapter 2

Background

This Background section includes details on the phenomena associated with the simulation/experiments that are of interest. As the use of numerical and computational methods in approaching engineering and science questions is growing with the constant advancement of computational technology the disconnect between the simulations/experiments that are done and why can easily grow due to the constraints imposed by specialization.

2.1. Fundamentals Boiling

Boiling is a process of phase change governed by the nature of the intermolecular forces of the medium of interest. As molecules of this medium of interest for phase change are of large quantities the potential between each molecule (i.e. the net attractive and repul-

sive forces) and each other molecule would be needed to be taken into account for an exact theory based on just those forces. The result from summing over all these numerous contributions is the concept behind partition functions where the Lennard-Jones potential is used as the potential energy for each individual molecule, which is often used as a starting point to understand these material dynamics at the level of molecular-bulk interplay before application specific corrections are applied.

Building up from fundemental constituent parts may work eventually, but it is rarely the best route. For this reason, a research must consider at what level there is need to inquire. Depending on the research concerns in a given system the method of reseaarch will become more evident as specific models for understanding a medium's nature have corrisponding levels of interest interaction and interst in mind. The vapor pressure, chemical potential, Gibbs free energy, or equation of state may be the descriptive model used to understand the mediums behavior (See section 2.1.3 for more on these items) however bulk measurement may be far more useful. With this in mind these following items help inform the use of simulation methods.

The primary motivation for the use of nucleate boiling in heat transfer engineering applications is the ideal heat flux for values of wall superheat in the range of the saturation temperature (see fig 2.1). For temperatures slightly above the saturation temperature large heat flux is found provided the boiling remains in the nucleate regime and does not cross over into film boiling. Nucleate boiling is when bubbles are actively forming and depart-

ing allowing for liquid to mostly remain in contact with the heat transferring substrate, whereas film boiling occurs when a layer of vapor prevents liquid contact with the substrate thus reducing the rate of heat flux for the same temperature.

Additionally, this near saturation temperature is maintained at the boundary provided values at or below Critical Heat Flux (CHF) are accommodated insofar as for slight increases of temperature there is a correspondingly much larger heat flux which may have a restoring capacity. CHF is a process where the maximum amount of heat flux between the interface and the boiling medium is taking place so that any more heat flux will cause a boiling crisis [119, 120] where the medium will surpass the Leidenfrost point. At this point, there is a layer of vapor between the interface of heat transfer and the boiling medium (composed of the boiling medium). This is because at the Leidenfrost point the heat transfer was more than nucleate boiling could maintain leading to a constant vapor or dry out of the surface. This in turn means that the surface has less than ideal heat flux characteristics. The previous combination of liquid dense liquid-conduction and bubble induced mixing is lost. Now vapor-convection and radiative processes account for all of the heat transfer. Proportionally the amount of heat transfer at a given temperature performance is also reduced. In a realistic system, energy (relating to heat) rather than temperature on its own is more likely to be the maintained parameter or run in excess for that matter. The interest in maintaining the maximum heat dissipation while also never risking CHF is why engineering methods to extend the CHF value for a given setup or

form a control a system to prevent CHF are areas of research interest.

Heat flux drops considerably outside of the nucleate boiling regime. This is in part explained by the changing characteristics of the boiling type (Nucleate changing to Film) however is this sufficient? The enthalpy of vaporization, a crucial factor in the heat is still present. This seems to suggest that the liquid contact and mixing effect of repeatedly forming and moving bubbles is also crucial to the enhancement nucleate boiling enjoys. That is the loss of conduction and mixing are responsible. This interpretation misses that: 1.) The rate of mass-phase change is NOT held constant. 2.) Any vapor convective component in the film boiling regime is unlike a normal circulating convective effects such that a radiative process of heat transfer is now the dominant path to the near exclusion of others. The heat transfer can be expected to follow the heat equation where the gap of vapor further degrades the performance.

If the heated surface, or wall super heat, becomes too much for the liquid to dissipate with nucleate boiling the boiling regime changes to film boiling. Film boiling cannot maintain the same level of heat flux that nucleate boiling can so in practical applications where energy (heat) is of concern rather than temperature this leads to the boiling crisis known as passing Critical Heat Flux (CHF). From here the wall superheat will become a

If there is a heat generating device that uses boiling to dissipate said heat, the runaway problem of film boiling is now evident, however, heat is not in endless supply. The reverse process also takes places and finds a sudden drastic increase in heat flux as the transition

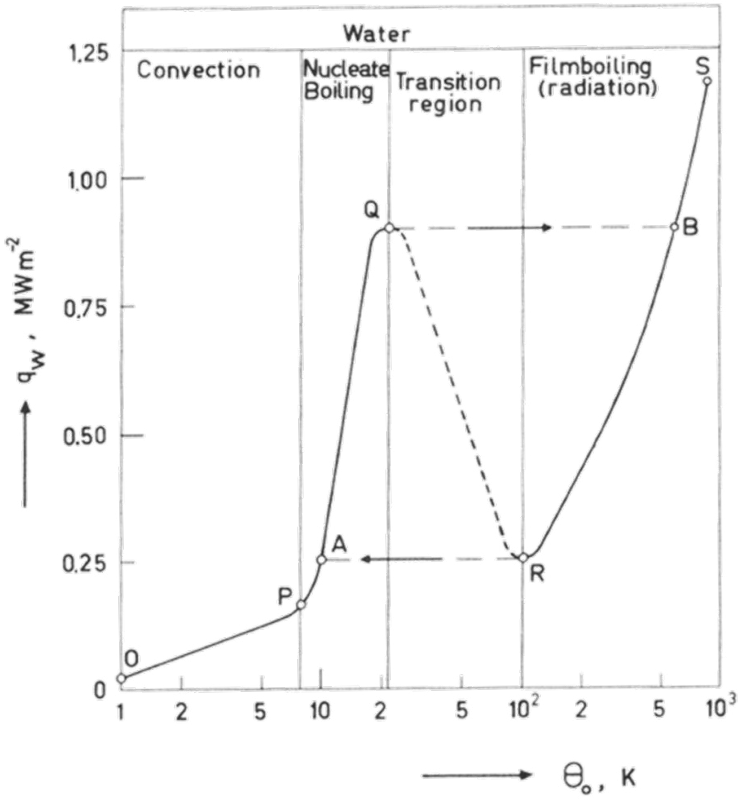

Figure 2.1: The Figure depicts the isobaric boiling curve. The heat flux q is given as a function of the heated surface (wall) temperature θ above the saturation temperature (wall superheat). Temperature is given logarithmically. Heat transfer is shown to be maximized with minimal temperature difference in the nucleate boiling regime PQ. For increasing temperature, point Q is CHF and R is the Leidenfrost point of stable film boiling. RS, requires over an order of magnitude increase in superheating for similar levels of heat flux to those of the nucleate boiling PQ [125].

out of film boiling occurs. Fig. 2.2 shows the CHF phenomena moving from film boiling to nucleate boiling with a droplet as the subject.

Progress in understanding more about the boiling process leaves even more areas of computational and experimental research available to pursue. Despite the growth in fields such as optics, instrumentation/detectors, and image processing, the boiling process still

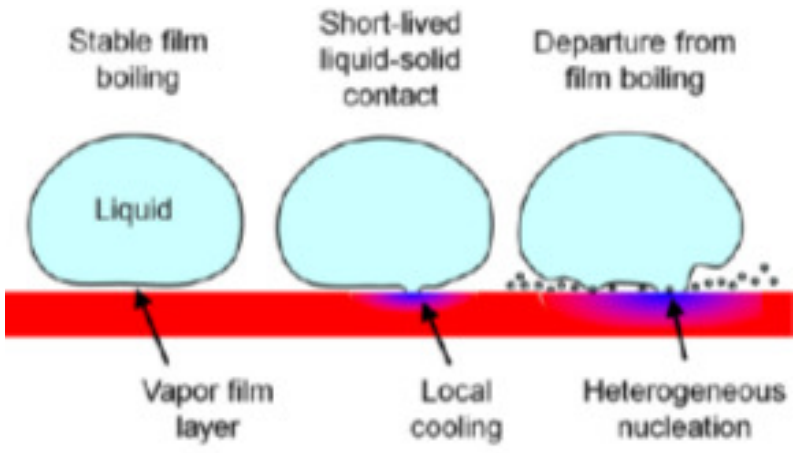

Figure 2.2: This figure displays the reverse of the CHF-Liedenfrost-film boiling transition. From left to right: a droplet in stable film boiling, once the heat flux is low enough the vapor can cool the heated surface enough for short-lived liquid contact which further cools it (center) until CHF is reached from the high temperature side (right) and nucleation begins [58].

holds secrets where techniques new and old are pursuing similar lines of inquiry [74].

Measuring phenomena associated with boiling and other two-phase heat transfer applications is made difficult by the multiple phase boundaries which are not set at perfect straight angles. These phases indexes of refraction can be to close or too far from unity for easy optical access. This is to say that if the index of refraction for water was identical to air it would permit perfect transparency, yet it would also be invisible. Conversely that the air and water vapor are very close in index of refraction, yet liquid water (1.3) is close to eyeglasses (1.5) means that the arrangement of bubble systems are akin to deciphering a series of lenses. Add to this that any further measurement interference with lab apparatus, electro-contact sensor probes, or acoustic devices having potential to cause changes in the underlying physics, and it is a difficult field of study at the microscale.

This is the benefit and hindrance of adding computational science to the loop of understanding the boiling process. In its benefit: the promise of access to the inaccessible and to its hindrance: and expectation of a leap from mechanistic correspondence akin to what computation has done for lithography.

2.1.1. Boiling Motivation

Boiling phenomenon is of great use for industrial processes as it is the highly effective way of transferring heat due to the enthalpy of vaporization [125]. Additional aspects of interest include controlled volumetric multiphase or rheological applications, and vapor generation. These include functional/additive printing, combustion, and power generation respectively. Increasing understanding of the fundamental mechanisms and leveraging the same are tied together with engineering in these pursuits as phenomenological research likewise feeds into the state of the technology. Established and leveraged mechanisms supported with computational research include: transport phenomena [83], microcavity/nucleation departure [64], and use of microchannels [51, 63].

There are different sub-regimes of boiling that are of consideration in bulk flows, however of the main divide, nucleate boiling vs film boiling, nucleate boiling is the one which is being discussed here because it shows a fully developed process of boiling that can be properly quantified offering the greatest rate of heat transfer at the lowest temperature. Furthermore, due to the phase change, the boundary temperature is es-

sentially constant for a boiling mediated heat transfer process (provided the heat flux is within an accommodatible range).

One other capacity that is offered in the use of a boiling process is the physical transport of the boiling medium in heat exchangers. Although flow boiling is an obvious example of this, pool boiling can accomplish a similar end. A heat exchange circuit of the boiling water is used for a thermonuclear power plant reactor core which is physically transported to the steam generating water. In computer/server cooling, a circuit of incoming water can be routed to the heatsinks for chip cooling [57] and is part of the ongoing International Roadmap for Devices and Systems (IRDS) [2]. Solar concentrators also move the medium (water) through the system [26, 121, 134]. Theory based experiments have lead to advances such as:

2.1.2. Condensation the Counter Part

This Subsection is offered almost as a vignette for perspective, particularly in how systems which include a boiling process often contain a condensing process though this may be overlooked as such by researchers from either perspective especially where only one phase change process in the system circuit is limiting or close to limiting.

The condensation process is useful in a variety of industrial applications. These range from gaining fresh water through desalination or directly from even seemingly dry air

sources. Cooling air for living in AC units or goods in refrigeration. Considering the primary topic of boiling addressed in this work, condensation can be the counterpart in an overall system phase change recycling system. Just as in others mentioned where condensation is optimized; condensation devices used where the boiling process is more likely to drive performance include open water reclaiming devices for steam operated units such as turbines or closed reclaiming devices such as where boiling is used to dissipate heat so maintaining the medium (e.g. water) near saturation temperature permits maximized rate of heat transfer potential for a given condition system.

In a similar if not analogous way to nucleate boiling having the ideal characteristics for the boiling process, dropwise condensation, where droplets form and then roll away due to gravity (orientation contingent). Dropwise condensation is energetically most ideal compared to film condensation. Surface functionalization can alter properties overall or in selective/patterned ways. The along with the thermal properties of a material used in condensing, wettability (the droplet contact angle) is often of primary concern as this will change the approximate characteristic departure size of a droplet. A water droplet on a vertical surface usually can grow to 3 mm before it loses equilibrium, but the larger or smaller sizes depending on specifics of the interface and therefore can be engineered for the conditions.

In further investigation promoting or inhibiting departure. and keeping the dry surface free also alter a system towards an engineered ideal. Understanding the exact temper-

ature distribution has been considered as a means to quantify fine differences, hence thermography, direct optical investigation of triple-phase contact line, and morphology. Miljkovic et al. [73] shows how the microsystem effects can be so powerful as to combine in somewhat surprisingly mobile droplets. These areas of investigation more broadly are all functional enhancements.

Functional enhancements, such as passive coating types that address the high surface energy of the materials used in heat exchanging devices are most common. Zhao and Burnside [140] reported an approximate heat transfer coefficent rate doubling with values of 6-8 kW $m^{-2}K^{-1}$ with the enhancement of Activated Reactive-Magnetron Sputtering Ion Plating (ARE-MSIP) technique for coating brass copper condenser tubes. Active enhancements can be electro-active, mechanical, chemical, electro-mechanical-chemical, or any permutation thereof [93, 142] but, of course, require added input. Unlike how dryout is considered universally a problem in boiling, film condensation and surface wicking are harnessed in some enhancements. An aspect common to active enhancement tends to be the clearing, which is less of an issue for the boiling process due to buoyancy, though advancements in pool boiling have used guided channels for the bubbles to induce a jet of mixing liquid over the heat sink [55]. Ahlers, Buck-Emden, and Bart [5] report on the practical implications and concerns as life-cycle maintaining dropwise condensation for energetically favorable reasons is a signifigant part of the consideration for advancing condensers, but this includes failure modes and

enhancements informed by the innovation cycle.

2.1.3. Statistical Mechanics Perspective

Statistical mechanics, intermolecular dynamics, and material science are useful in informing an understanding what principles are driving boiling phenomena as well as fluid dynamics more generally. There is a significant distinction for the computational considerations that will be made, this is that each species or phase of a multiphase simulation is not simulated from first principles nor does an sort of loosely atomic or elementary parton theory apply. To be clear there is no tiny piece of liquid nor a little bit of vapor. Pressure is not due to bits collectively rebounding off of a surface. Temperature and kinetic energy have no relation in these expressions as they perhaps *should* strictly speaking. The phases are representing the bulk flows. The interest is the net current, the flux, but not the atom, molecule or simulation corpuscles because the former is sufficient to satisfy the Navier-Stokes numerical solver.

When considering intermolecular models, often a potential that is used, which is the combination of the attractive and repulsive forces between any two molecules, to relate to the energy of the two-body problem is the Lennard-Jones potential. In this model the attractive force is due to the Coulombic interaction and found in higher order multiple perturbations of quantum mechanics such as van der Waals. The repulsive force includes nuclear scattering from the strong force, Pauli exclusion, and the impact of

the total inter-atomic/molecular angular momentum.

In a non-trivial way the exceed this purpose, a partition function can be formed for the set of all molecules leading to the common Equation of State results which are shown to be contingent on the intermolecular potential.

The heat of vaporization is the energy taken up with to do the work displacing volume (atmosphere) and dissociating the chemical potential (liquid bonding) thus greatly increasing intermolecular spacing.

Integration over the Lennard-Jones potential by the mean increased spacing (vapor-liquid density change). The latter is usually the main contribution, and for water at Standard Temperature Pressure (STP) it is just over 91% [107].

2.1.4. Boiling Phenomenology

There is an ongoing interchange of research across disciplines and degree of application towards discovery of deeper understanding of phenomenology and the development of new or improved technology. In terms of technological advancement this is the cycle of innovation. This inclusive space of human investigation is broad including at the extreme those that strive to study nature for the wonder and beauty of nature alone. They are served immensely by the advances in technology to permit these investigations. Likewise, but in a more circuitous path, those on the other extreme that are

concerned with technological advancement alone without any concern for deeper understanding of the nature that governs technology's processes will inevitably be served by the advancements of the previously mentioned group.

Of course this dichotomy is offered as an extreme to illustrate the point, though such extreme personalities are rare, the realistic cases show this interplay. The researchers in the astrophysics of gravitational waves are also keenly focused on instrumentation and detectors. Such as the 2015 evidence of gravity waves, Laser Interferometer Gravitational-Wave Observatory (LIGO) [19].

The histograms with credible bounding show how detection is translated into a picture of the underlying natural event [1]. Simply put, the astronomer needs the ever advancing telescope technology. Finishing the theme, device manufacturing is mostly about throughput not deeper understanding. The simple transistor and its constant progress through improved phenomenological understanding is what drives many technological manufacturing goals; everything from cabinetry to software to the direct lithography of ever reduced pitch. In this example supporting technology is considered, but for a hard figure, the USA Semiconductor Industry Association (SIA) reports approximately one fifth revenue reinvestment to R&D in a 265 billion dollar and rising industry that also sits as a top 5 export good [3, 4].

Below are some phenomenological interests related to this body of research. These serve as motives but also in this research-interplay as tools in further advancing the

state of the art. Of particular interest is what dictates the type of simulations that will be most useful. Notice references to experimental sections due to these items being used in this body of research to a degree.

The Microlayer:

The microlayer is perhaps the most widely applicable part of bubble anatomy in terms of encapsulating the demands and interests of investigating nucleate boiling. It is beneficial for measurement, advancing theory, historical context, and testing computational realism. The microlayer is located beneath a bubble in nucleate boiling. It is a thin film thickening concentrically with the bubble's circumference as viewed from above (see fig.2.3).

In 1704 Newton published on what are now called Newton's rings [74] (the effect was reported previously by Hooke [39]). These rings (or lines depending on the orientation) are a type of visible optical interference pattern effect that indirectly yield thickness and thereby contour, if within range, of thin films. This effect was noted under growing bubbles (see fig.2.4) which led to the investigation of the microlayer.

In 1964 Sharp [104] cited Bankoff, Colahan, and Bartz [9] as the motivation for experimental investigation of the microlayer in the boiling process. In Sharp's experiment the microlayer was observed and measured. Cooper and Lloyd [20] found relatively simple theoretical calculations to agree with observations of this and similar work. The

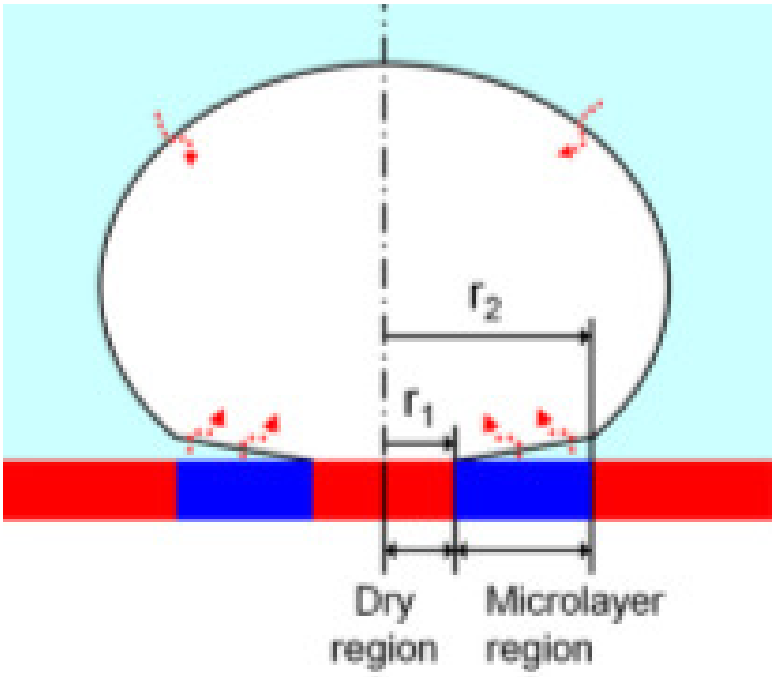

Figure 2.3: A schematic of the profile of a single nucleating bubble in a boiling process with the microlayer depicted above the blue region (labeled) [58].

theory tended toward the finding that the majority of bubble growth comes from microlayer evaporation. Cooper and Lloyd used an unspecified computer program with the capacity to add evaporation or condensation to the top of the bubble [20]. Conclusions at the time for factors that encourage microlayer formation were high superheat, low subcooling, and low ambient pressure. Jawurek produced photographs of an underneath view based interferometry of the microlayer with simultaneous bulk boiling profile [78]. Jawurek's finding was that the microlayer thickness increased near the outer edge with continued heat transfer while the center dried to a characteristic radius and stopped. From here many different investigations into the microlayer of various fluids have been undertaken.

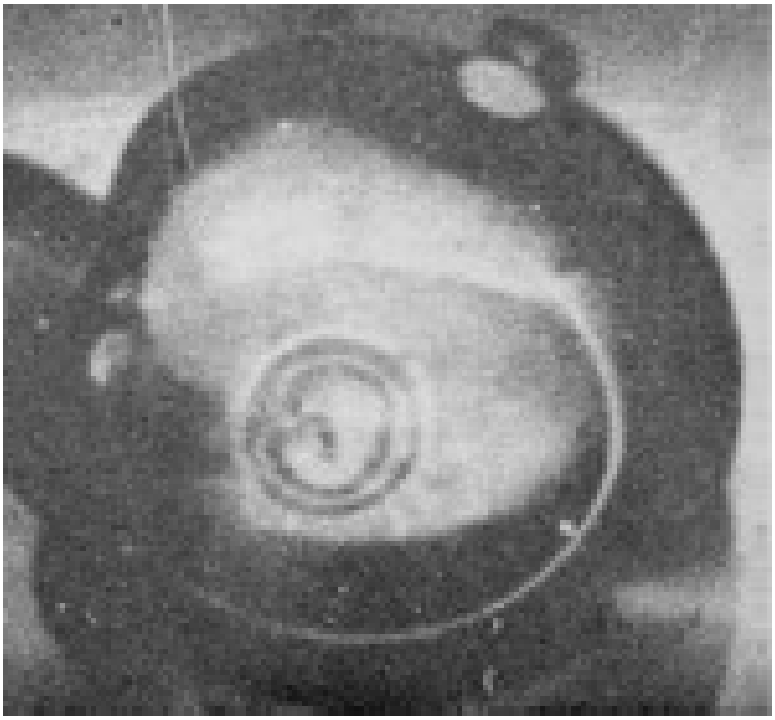

Figure 2.4: A photograph of an interference pattern created by a bubble's microlayer. The pattern, Newton's rings, can be used to measure thickness [104].

The microlayer is discussed further in context of the heated plate experiments/simulations performed in section 3.4.1.

The Thermal Layer:

The mass flux of a growing bubble is dependent on the interfacial temperature gradient at the interface. Scriven derives a theory of bubble growth [101]. The interaction between the liquid and the interface creates a thermal film in the liquid adjacent to

the interface. The radial depth of this thermal film increases with the time evolution due to bubble-liquid conduction heat transfer. This thermal layer is thin enough that sharp interface methods are needed to simulate its development, though this is a minimal standard only. As the thermal layer develops the interfacial temperature gradient continues to change causing a interaction loop.

The Thermal Layer/film is discussed in context of the non-adiabatic experiments/simulations in section 3.4.

Triple Phase Lines/Points:

Aktinol et al. [7] continue the advancement of simulation methods for multipase flows noting that the heating wall is all but exclusively a boundary condition. Use of finite difference methods for the inclusion of the wall inter-variability for increased success in triple phase points/lines. The phenomena of triple-phase contact lines can be of benefit in studying multi-phase heat transfer. These lines are present in nucleate boiling (near CHF and dry out), dropwise condensation/coalescence, and quenching processes.

The near dry out/CHF contact line effects of note in boiling may be considered a distinct boundary layer forming a network of contact lines prior to the film transition as proposed by Theofanous, Dinh, Tu, and Dinh [119, 120] and disputed buy Chung and No [18]. This relatively important region is very difficult to model well.

In dropwise condensation the maximum heat flux occurs at the triple-phase contact

line [58]. Dropwise condensation requires the condensing surface is not wetted by the condensate [93]. This is the motivation for the functional enhancements discussed in section 2.1.1.

Quenching occurs when a wall (for convention, but any item) superheated above the Leidenfrost point is brought into contact with liquid. As the Wall cools the transition from film boiling to nucleate boiling is unstable and likely driven by local contact points/lines which provide quick disruptive vapor bursts from nucleate boiling.

Triple phase contact lines are discussed in context of the heated plate experiments/simulations in section 3.4.1.

Stable Flow in Microchannels

Kandlikar [13, 50, 52, 54, 55] has reported extensively on microchannel use for heat transfer in liquid cooling applications. This is a subtopic of both flow boiling and microstructures/enhancements. This field is being focused on due to related pertinent avenues of investigation for simulation and to avoid generalizations. Pool boiling is the process wherein heat transfer is supplied to the heat sink by immersion in a pool of the medium (water or coolant). Flow boiling uses a constant flow of the medium across the heat sink. The added complication of providing a flow increases implementation and maintenance difficulty for flow boiling, however increased heat flux has been an incentive for continued research. Moreover flow boiling offers cooling in places where

pool boiling would not have physical Clearance. This is because the cooling process is physically transported with the medium. In the macroscopic heat exchanger this relocation is easily overlooked, however as considerations for higher density IC chips become realized with layered components, wall cooling may become insufficient.

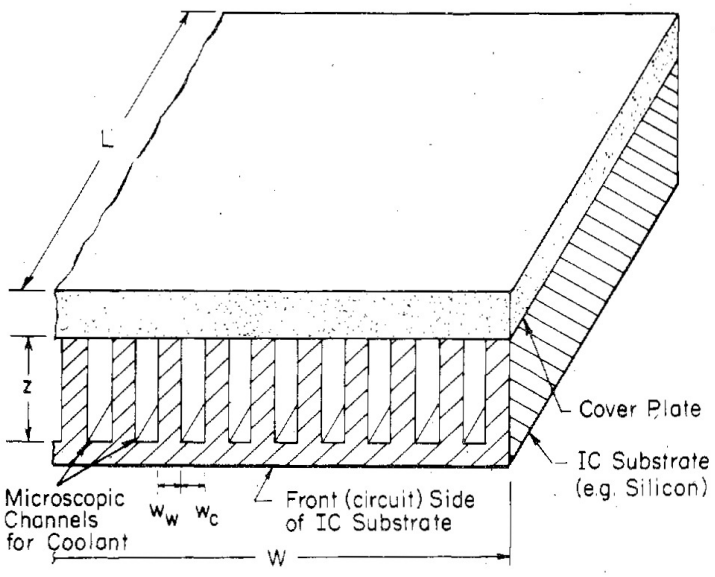

Figure 2.5: A schematic from Tuckerman and Pease originating article, the dimmensions of z and both w are 365 and 57 microns respectively. This design is shown as silicon with a philosophy to integrate the flow cooling as part of the IC circuit [123].

The use of microchannel array heat sinks (see fig2.5) is attributed to Tuckerman and Pease [123] who's design was an integrated component for IC circuits and able to dissipated 790 W/cm^2 with $71°C$ temperature difference.

The phenomenology of interest when benchmarking is no longer just monitoring the CHF transition and Heat Transfer Coefficient (HTC), but rather maintaining stable flow for a flow boiling regime. This is because as higher heat fluxes are achieved the nu-

cleating bubbles tend to impeding the flow channel. Stienkie and kandlikar synthizied where flow losses are introduced. [116]. Instabilities within microchannels are the precursor to local CHF and a predictor of channel end performance. Despite providing nucleation sites, the lack of depth in microchannels allows for conflicting nucleation which is one of the factors present in microchannel flow instability. Since the problem is due to a number of factors manufacturing/measurement uncertainty/precision, pressure drop, proper choice of dimensions, inlet/outlet management, and particular medium there are many items on account which may offer added stochastics.

For these above considered phenomenon of the microlayer, thermal layer, triple phase contact lines, and technonolgical advancement in microchannels it is clear:

1. Methods for simulation without interface details that instead rely on making end point analysis based on existing systems cannot extend information about interface interactions in systems beside those from which they gather their data.
2. Methods that do define interfaces but have use a smeared interface (many: IR, VG) to prevent errors as time evolves will have only a smeared image of these interfaces, and are unable to resolve interface details (thermal layer).
3. Methods with high interface detail but without mass transfer accounted for strictly will not be helpful in both answering questions of mass flux and interface tracking that is dependent on mass flux.

Conclusion: for these reasons, choosing a simulation type capable of sharp detailed interfaces and exact mass transfer accounting is best suited for of advancing understanding in microscale phenomena.

2.2. Interface Tracking Algorithms

Numerical simulations of multiphase flows are challenged by the abrupt discontinuity that is the interface of the phases. The characteristics on either side of the interface may vary drastically and the interface itself may also have particular physical characteristics to implement. Although there can be endless methods to create such simulations ranging from those that are very realistic to black-box-experiment-esque (in that little to no detail but a final solution is ascertained or only boundary conditions are simulated for improving a component within a system). Details in simulations are driven by the information needs of the user/designer as extraneous details add implementation and computational time and expense, however without some buffer it is possible to neglect details which have an impact. Apart from systems driven with a restoring force to particular modes, small changes can have a lasting impact on final conditions. Most simulation methods in the research area near nucleate boiling include information along time steps and some resolution of the domain being simulated, but the realism is at the same time reduced by assumptions made to the process such as the phases are rarely composed of realistically interacting molecules and pressure is a characteristic not a result of these molecules

imparted impact.

The explored space of numerical simulations of multiphase flows divides mostly along the Eulerian, Lagrangian, or in certain perspectives a combination of both of these two method types. This broad division of type has to do with how the discrete computational units or cells are oriented and organized. Computational cells of an even size set in a regular array determined by the coordinate system are of the Eulerian Method. Interface-centric cells of variable size are of the Lagrangian Method. While Lagrangian methods can be simply thought of as simulations where the computational mesh is moved to describe the interface as time evolves. The Eulerian methods are those of a fixed mesh where the variables of each element in the mesh are changed to express the simulated physics. The following subsections give a brief overview of available interface tracking techniques.

2.2.1. The LS interface tracking method

Osher and Sethian [77] proposed the Level-set (LS) method. The LS method utilizes Eq. (2.1) to track the interface.

$$\frac{\partial \phi}{\partial t} + \bigtriangledown \phi \cdot \overrightarrow{v_{int}} = 0 \tag{2.1}$$

where ϕ is the distance from the computational cell to the interface and v_{int} is the

velocity of the interface.

Figure 2.6 shows the values of ϕ in a computational domain with a spherical bubble as reported by Sharma [103] while employing the LS method. It is observed that the LS values are zero at the interface, positive at phase-1, and negative at phase-2. The magnitude of the LS function at each computational cell increases with the distance from the location of the cell center to the interface. Also, to avoid numerical instabilities, it is common to define regions near the interface in both phases. The transition region utilizes functions (e.g., the Heaviside function) to define a smooth change in the fluid properties.

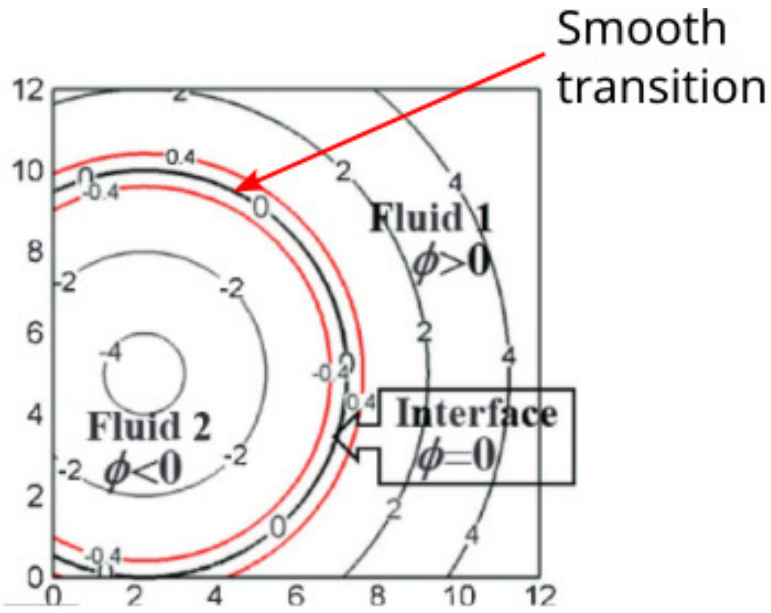

Figure 2.6: values of the ϕ in the LS interface tracking equation for a circular interface. A smooth transition across the interface defines the fluid properties. Figure adapted from [103].

The LS utilizes the interface velocity to find the change in ϕ at the computational

cells. This implies that for each time step, all the computational cells should adjust their numerical values based on the interface velocity and location. The LS method requires re-initializing the values of ϕ after a certain simulation time, which may lead to issues with mass conservation. Son indicated adding a volume-correction step to the LS formulation to ensure satisfying mass conservation [112].

One of the main advantages of the LS method lies in the smooth transition of the LS function across the interface. Such a smooth transition allows an accurate estimation of the gradients of the LS function required for proper modeling of surface tension based on the interface normal vectors.

2.2.2. The VOF interface tracking method

Hirt and Nichols [41] proposed the volume-of-fluid (VOF) method to track interfaces ensuring mass conservation. Assuming null mass transfer between phases, the VOF method utilizes Eq. (2.2) to track the interface.

$$\rho_1 \left(\frac{\partial \alpha_1}{\partial t} + \bigtriangledown \cdot \alpha_1 \overrightarrow{v_1} \right) = 0 \tag{2.2}$$

where ρ_1 is the density of the vapor, α_1 is the phase-1 volume-fraction, and v_1 is the velocity of the vapor phase-1. Perez-Raya and Kandlikar [82] explain the fundamental principles of the VOF gradient method applied to 1D models. Figure 2.7 shows the

values of α_1 as reported by Garoosi and Hooman [31] in modeling multiphase interactions with the VOF method. Results show that the VOF method ignores the distance from the computational cells to the interface. Instead, the VOF method looks at the volume fraction of the tracked phase. In the figure, cells with α_1 equal to one consist of phase-1, cells with α_1 equal to zero consist of phase-2 cells, and cells with $0 < \alpha_1 < 1$ consist of interface cells. The VOF method focuses on the cells that have an interface to perform the interface tracking.

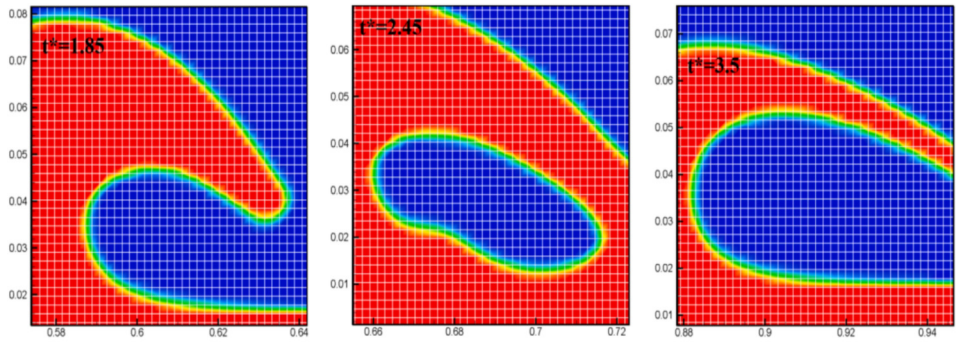

Figure 2.7: VOF contours of a multiphase flow at three different instants. Figure adapted from [31].

One of the main challenges in adopting the VOF method arises from the sharp discontinuity of the VOF function at the interface. The sharp discontinuity leads to numerical errors in estimating the normal vectors required to determine the change in the volume fractions at each time step. Errors in the normal vector adversely affect the modeling of surface tension, which depends on the normal vector. The errors translate into parasitic velocities that can generate adverse interface deformations. Also, difficulties appear in advecting the α_1 values in a way that avoids interface breaking or interface

deformations.

Figure 2.8 shows the two main approaches to reconstructing the interface (i) donor-acceptor and (ii) piecewise-linear (PL). Hirt and Nichols [41] describe the donor-acceptor approach. In the donor-acceptor, the interface is assumed to be planar without an inclination angle, and the discretization schemes of the neighboring cells (including fluid properties) are modified to account for the location of the planar interfaces. Youngs [139] proposed the piecewise-linear approach, which assumes the interface within each computational cell to be a straight line. The approximation of a straight interface in the piecewise-linear approach gives the advantage of improved accuracy in representing thin liquid films that are a fraction of a cell thick, which is relevant in the modeling of multiphase flows interacting with solid surfaces.

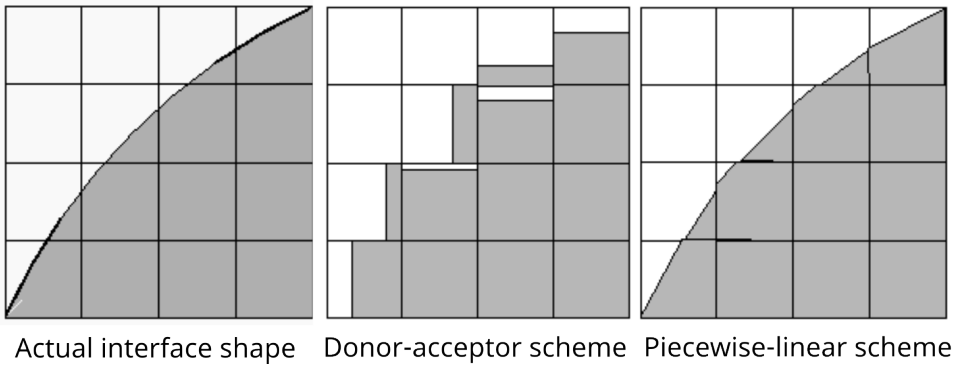

Figure 2.8: Interface reconstruction approaches in the VOF interface tracking method.

2.2.3. The coupled LS-VOF interface tracking method

The coupled level-set volume-of-fluid (CLSVOF) interface tracking method was proposed by Sussman and Puckett [117]. CLSVOF eliminates adverse interface deformations with accurate computations of the gradients of the LS function and conserves mass by advecting the interface with VOF interfacial values. As a result, CLSVOF method may perform better in multiphase modeling experiencing strong deformations. Fig. 2.9 shows results reported by Dianat et al. [25] in the modeling of a sliding grooved surface. Results indicated that simulations with the VOF method lead to interface smearing when the droplet moves along the groove, and simulations with the CLSVOF method can preserve the interface shape. The difficulties in achieving a successful coupling and additional considerations needed for modeling phase-change conditions represent the main disadvantages of the CLSVOF method. Also, there exist a diverse number of applications of multiphase flow modeling that are not subject to strong interface deformations.

2.2.4. The particle tracking interface tracking method

The interface marker particle method is attributed to Daly [21, 22]. Interface marker particle methods utilize the Lagrangian concept of following individual particles with the Eulerian fixed cells. Fig. 2.10 shows the location of the particles on a 2D circular interface. The interface position is determined with the Lagrangian particles which

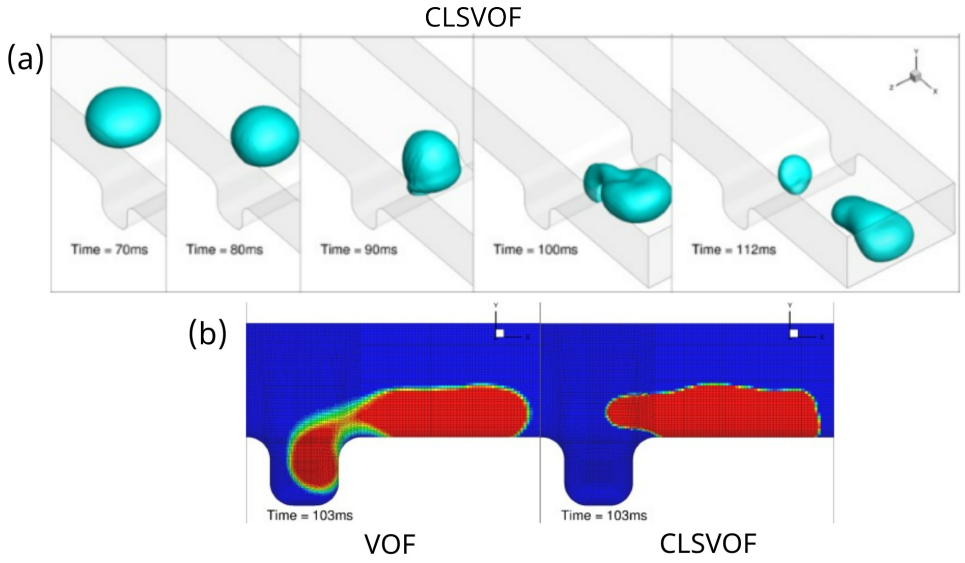

Figure 2.9: (a) 3D simulation of a sliding water droplet through a grooved surface, (b) 2D contours of volume-fraction at a specific instance in the sliding droplet obtained with the VOF and CLSVOF methods. Figure adapted from [25].

are capable of quality output, but a tradeoff must be made between too many particles where it becomes computationally expensive and misinterpretation can result in surface tension, extinction; or too few leading to error in the interface and reduced mass conservation (potentially). Many instances will interpolate the Eulerian volume fraction with interface details [87]. Juric and Tryggvason [49] use a similar method to simulate film boiling. The drastic changes in the curve of the interface were modeled without degrading fluid velocities. Despite the technical care needed and computational demand, Van Sint Annaland [124] shows that this remains a promising method, for the proper circumstances, here with 3D bubbles rising, but not mass transfer.

The contrast of the lagrangian type movable-mesh interface marker particles requires

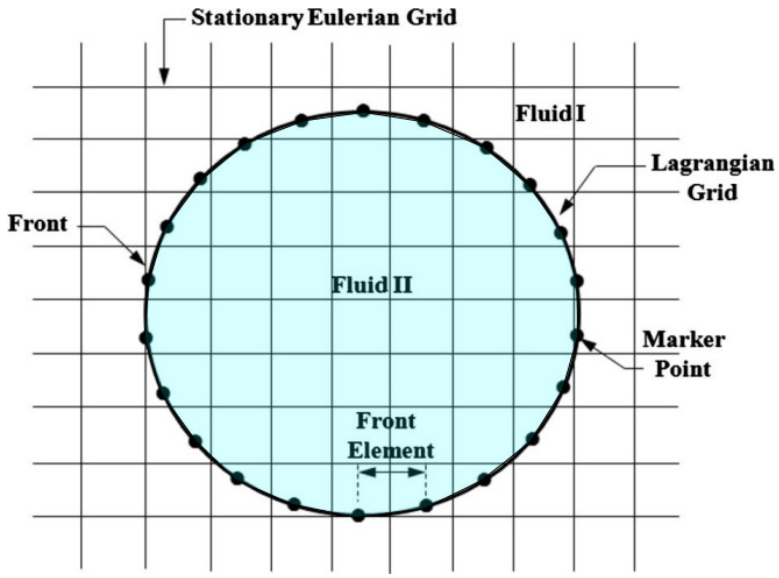

Figure 2.10: Front tracking method utilizing markers to advect the interface in a Lagrangian approach. Figure adapted from [46].

some stage of simulation coupling which increases computational expense.

The Computational particle-fluid dynamics (CPFD) method introduced by Snider [109] is a method where the gases or vapor component is treated in the Eulerian regime and the liquid or particulate is treated in the Lagrangian regime. The CPFD method is of interest for mixing applications in the fluidized/gassified beds sector [66], for chemical reactor study in biomass [70, 95] and this method is applied to related liquid injecting industrial applications [86].

2.3. Mass Transfer Models

Numerical simulations divide the computational domain into multiple computational cells (see Figure 2.11(c)). In computer modeling of multiphase flows with two phases, these computational cells are classified as phase-1 cells (cells filled with phase-1), phase-2 cells (cells filled with phase-2), or interface cells (cells that contain the two phases divided by an interface). The cells that lie next to interface cells are known as neighboring cells.

The simulation should estimate the mass transfer corresponding to each interface cell (amount of phase-1 transformed into phase-2 or vice-versa). For instance, Figure 2.11(a) shows a growing bubble immersed in liquid in contact with a heated plate. The bubble grows due to the evaporation of liquid at the interface; the amount of liquid that evaporates at the interface is termed interfacial mass transfer.

The literature distinguishes three methods to compute mass transfer shown in Figure 2.12, (1) empirical coefficients, (2) VOF gradient to find the interface size, and (3) interface reconstruction. The following subsections give a brief description of the three main methods.

2.3.1. The Empirical Coefficients Method

Table 2.1 shows the publications applying the Empirical Coefficients Method (ECM) consisting of about 37% (10 publications) in the field of boiling modeling from 2018

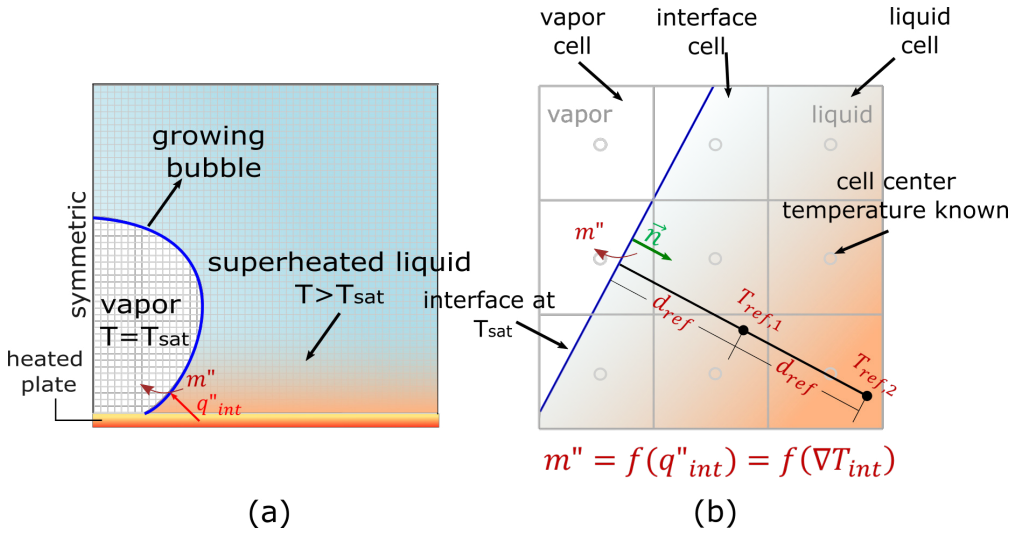

Figure 2.11: Need to compute mass transfer in a simulation of boiling: (a) computational domain with a bubble, and (b) definition of cells and estimation of mass transfer with temperature gradients.

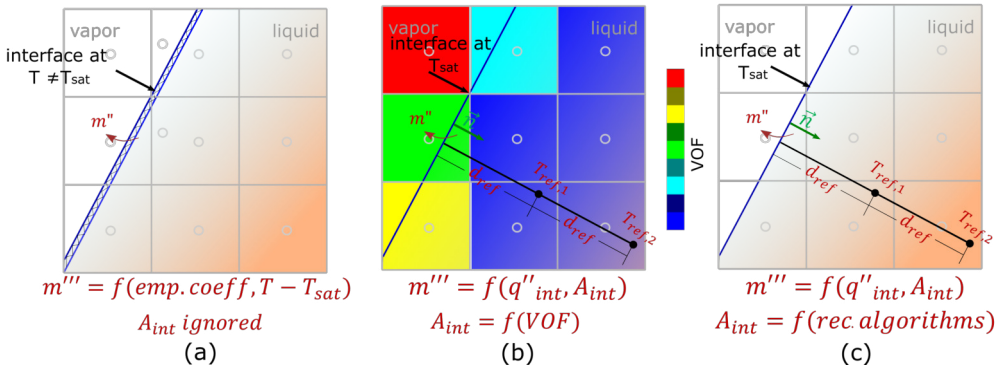

Figure 2.12: Methods to compute mass transfer (a) empirical coefficients, (b) VOF gradient, and (c) interface reconstruction.

to 2022. The empirical coefficient method uses empirical factors and assumes an interfacial temperature difference to compute the mass transfer. From an analytical or symbolic perspective, the ECM determines mass transfer independent of interface surface area see Eq.(2.3). Which by extension prevents accurate accounting of heat flux across the interface. This in turn prevents any adaptability to a system with an interface

that is deformed or changed in any way from that interface it was originally tuned with when the coefficients were determined.

The concept in play is that empirical measurement accounts for the mean interfacial area in a given domain to be found over the series of calculated computational cells, however, as a result extracting interfacial details such as heat flux is likewise subsumed into these coefficients. In principle the empirical coefficients method is commendable for its connection to experimental findings, data, its practicality for particular cases, and its theoretical elegance. Yet in practice, it can suffer from only having one term rather than a series of terms to account for the appropriate functional dependence at work. Much like the established theory of a series solutions in differential equations, where by nearly any sufficient number of terms may be used as a related rate contour solution, the ECM is a type of boundary value problem solution with limited flexibility from an experimental contour. In this way the ECM has the potential for highly practical results but a limited domain of viable application considering the potential for non-stable contours given the extrema of phase-change.

$$m'''_{\mathrm{lv}} = \mathrm{coeff}\ \sqrt{\frac{M}{2\pi \mathrm{RT}_{\mathrm{sat}}}}\mathrm{L}\left(\frac{\rho_g \rho_l}{\rho_l - \rho_g}\right)\frac{(T - T_{\mathrm{sat}})}{T_{\mathrm{sat}}} \tag{2.3}$$

where $coeff$ is an evaporation coefficient, M is the vapor molecular weight, R is the universal gas constant, L is the latent heat, ρ_l is the liquid density, ρ_g is the gas density,

Table 2.1: This table contains information on recently published pertinent research/research groups' articles that use the Empirical Coefficients method of calculation.

Articles Using the Empirical Coefficients Method			
Year	Author(s)	Simulation Details	Interfacial Thermal Behavior
2022	Kumar and Das [59]	2D-axisymmetric film boiling of liquid N2 (Note: VOF Gradient, and $m_{lv}^{'''}$ with EC)	Good, but not contact line
2022	Lee et al. [62]	3D flow boiling in microgravity	Not focused
2022	Yeo and Lee [137]	2D flow boiling in microchannels	Not given
2022	Yi et al. [138]	2D-axisymmetric nucleate boiling in microgravity boiling at microgravity	Thermal Smeared across interface
2022	Cao and Macian-Juan [15]	3D vapor bubbles condensation and effect of fluid properties	Not given
2022	Ha et al. [37]	2D-axisymmetric, condensing flows of single bubble and steam-jet flows of single bubble and steam-jet	Not displayed
2022	Bhuva et al. [11]	3D bubble coalescence in microchannels coalescence in microchannels	Not close to the interface
2021	Broughton and Joshi [12]	3D, flow boiling in microchannel	Smeared across interface
2021	Li et al. [65]	2D-axisymmetric effect of microlayer evaporation	Not displayed

T_{sat} is the saturation temperature, T is the temperature of the liquid-vapor interface, and $m_{\mathrm{lv}}^{'''}$ is the volumetric mass rate.

Juric and Tryggvason [49] simulated film boiling with the interfacial-resistance model; the authors made an entropy balance across the interface to find the temperature of the interface. Hardt and Wondra [38] pointed out that the interfacial-resistance model

applies to cases where the interface lies at a submicron distance from a surface, and errors may appear as the distance becomes larger. Figure 2.13 shows the reported bubble growth rates obtained with the empirical coefficients method. Results show that the model is not sensitive to the grid cell size, which allows ignoring mesh sensitivity analyses. In general, the model generates experimental trends observed in the growth of spherical bubbles immersed in superheated liquids. Figure 2.13(c) shows the effect of the empirical coefficient L on the bubble growth rate and its comparison with experimental results as reported by Cao and Macián-Juan [15]. Results indicate that the bubble growth rate is significantly affected by L, and only a value of L is 1.5×10^5 s 1 agrees well with the experimental data (the relative error is less than 15%).

Figure 2.14 shows the thermal and fluid dynamic behavior obtained by computer simulations adopting the empirical coefficients method. Results in Figure 2.14(a) show an ability to properly capture the decay of the temperature value to the saturation value at the interface; however, velocity distributions lack a velocity jump at the interface that should arise due to the liquid transformation into vapor. Figure 2.14(b) shows a thermal behavior that neglects the condition of the interface remaining at the saturation temperature. Results in Figure 2.14(c) show a nucleating bubble where the interface temperature differs from the saturation value, and the streamlines do not show a discontinuous jump at the interface. Figure 2.14(d) shows the temperature contours of simulating flow boiling; the temperature contours ignore the saturation condition at the

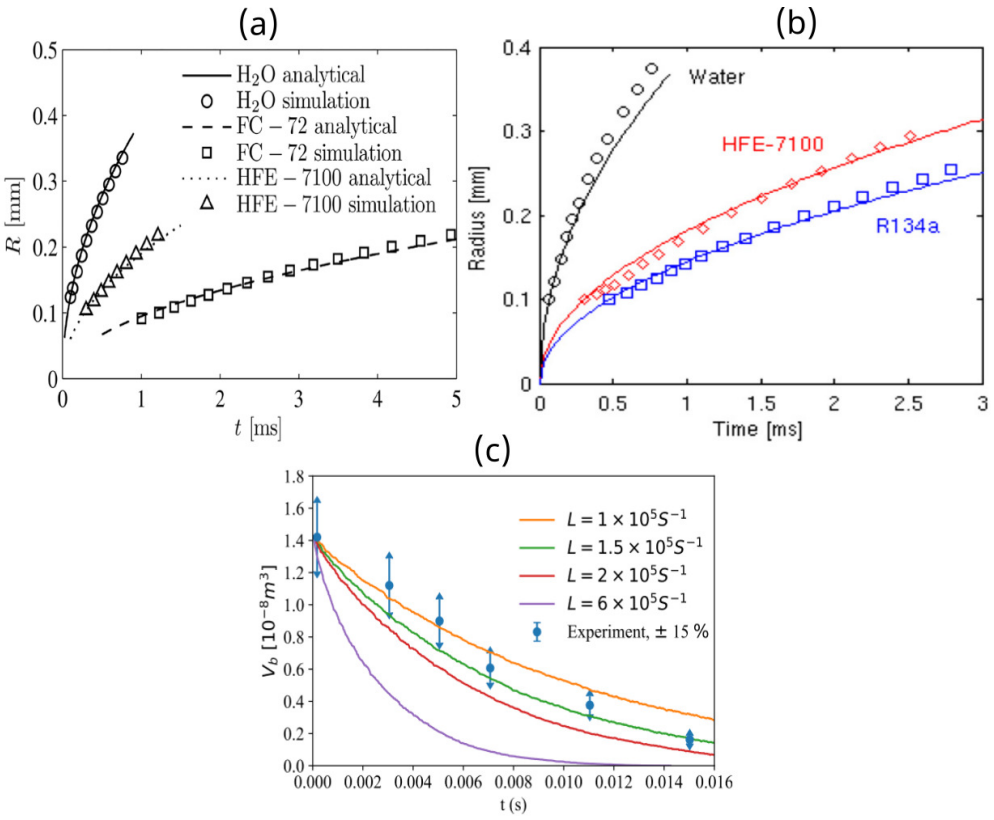

Figure 2.13: Bubble growth in superheated liquids. Comparison of simulations with the empirical method and theoretical solutions. Adapted from (a) Kunkelmann and Stephan [61], (b) Magnini adapted from [72], (c) and Cao and Macián-Juan [15]

interface, which may lead to deficiencies in capturing the heat transfer mechanisms.

Various simulations of boiling have used the interfacial resistance-model [38, 61, 72]

[48, 69, 135, 136]

However, as pointed out by Hardt and Wondra [38], the interfacial-resistance model applies to cases where the interface lies at a submicron distance from a surface, and errors may appear as the distance becomes larger.

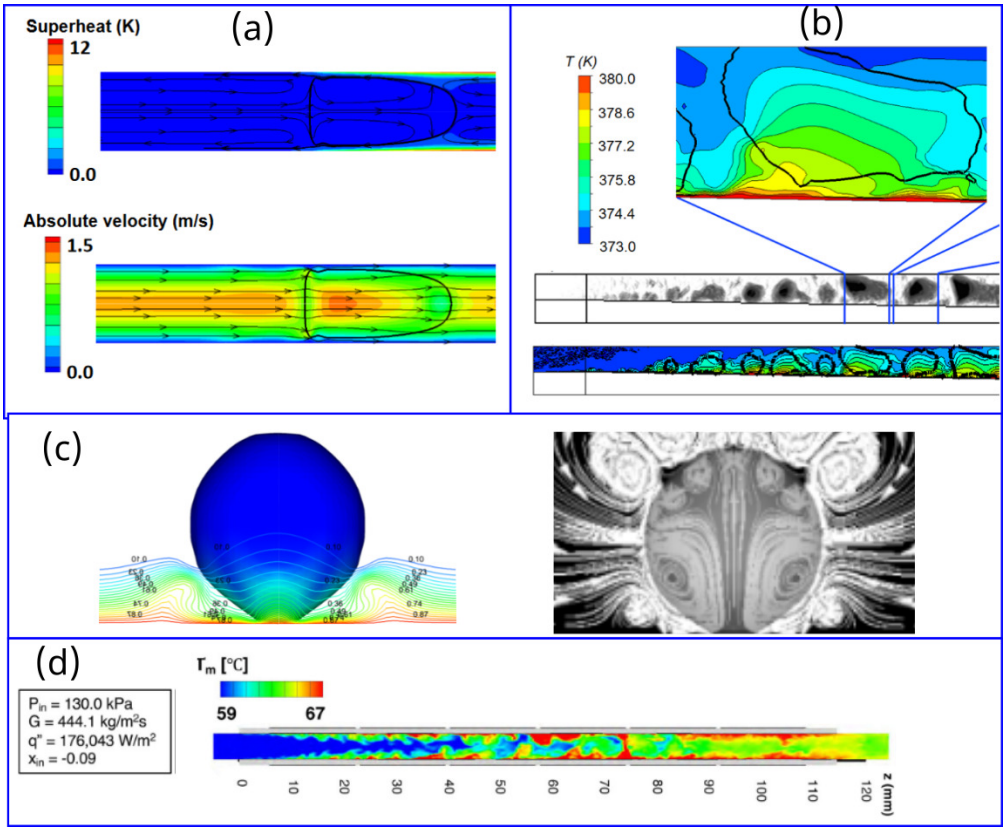

Figure 2.14: Temperature and velocities obtained by computer modeling adopting empirical coefficient method. Adapted from (a) Pan et al. [80], (b) Broughton and Joshi [12], (c) Yazdani et al. [136], and (d) Lee et al. [62].

2.3.2. The VOF Gradient Method

Table 2.2 shows the publications applying the VOF gradient method consisting of about 26% (7 publications) in the field of boiling modeling from 2018 to 2022. The VOF gradient method estimates the interface surface area by computing the gradient of the volume-of-fluid.

The interface surface area is computed Eq. (2.4).

Table 2.2: This table contains information on recently published pertinent research/research groups' articles that use the VOF Gradient method of calculation.

Articles Using the VOF Gradient Method			
Year	Author(s)	Simulation Details	Interfacial Thermal Behavior
2022	Shang et al. [102]	2D-axisymmetric, evaporating droplets on flat surfaces	Thermal distribution given
2022	Kumar and Das [59]	2D-axisymmetric film boiling of liquid N2 (Note: VOF Gradient, and m'''_{lv} with EC)	Good, but not contact line
2020	Fostiropoulos et al. [28]	2D-axisymmetric breakup of droplets with a boiling vapor bubble	Sharp interface temperatures given

$$A_{int} = |\nabla \alpha_v| V_{cell} \tag{2.4}$$

The VOF gradient method estimates the volumetric mass transfer with Eq. (2.5).

$$m'''_{lv} = m''_{lv} \frac{A_{int}}{V_{cell}} \tag{2.5}$$

Where m'''_{lv} is the mass flux, A_{int} is the interface surface area, and V_{cell} is the cell volume that contains the interface. The mass flux computed as a function of the temperature gradient at the interface and the latent heat of vaporization.

Figure 2.15 compares simulated evaporation/condensation rates obtained with the VOF gradient method against theory and experiments. Fig. 2.15(a) shows the results re-

ported by Shang et al. [102] of a mesh sensitivity analysis in modeling a condensing spherical droplet in a 1 s interval and with a slight change in the scaled diameter (from 1 to 0.9996); results indicate that simulations with the VOF gradient method can identify an optimal grid cell size in droplets condensing at a small rate leading to a minimal change in the droplet diameter. Fig. 2.15(b) shows the results reported by Shang et al. [102] comparing the condensation rate against experiments in spherical droplets (scaled diameter varies from 1 to 0.75 in a 1.5 s interval); the droplet in 2.15(b) condenses at a faster rate and the simulation showed good agreement against available experiments. Fostiropoulos et al. [28] reported the results in Fig. 2.15(c) in simulating spherical bubble growth; the results indicate that simulations with the VOF gradient method can reproduce theoretical trends with minor deviations (the study ignored the effect of the grid cell size). Please note the vertical blue line notes a break up time for the oil so the theory departing trend is not meaningful.

Figure 2.16 shows temperature contours obtained with numerical simulations adopting the VOF gradient method. Fig. 2.16(a) shows results reported by Shang et al. [102] in modeling an evaporative droplet suspended in a forced convective flow on a glass sphere (the diameter of the sphere is 0.4 mm). Results show temperature decaying to the saturation value at the interface indicating the ability of the model to identify interfacial heat transfer mechanisms. Fig. 2.16(b) shows temperature contours reported by Fostiropoulos et al. [28] in modeling oil droplet breakup. Results show the influ-

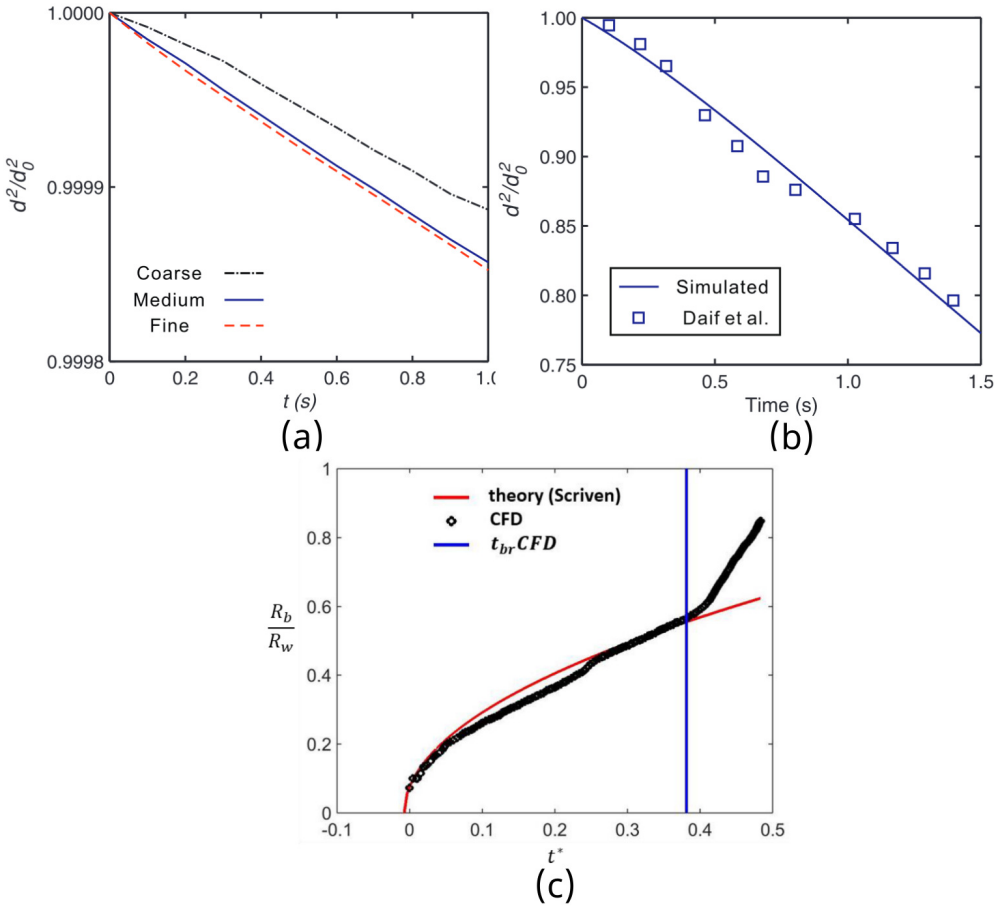

Figure 2.15: Condensation/growth rates obtained by computer modeling of spherical droplets/bubbles adopting the VOF gradient method. Figures (a) and (b) adapted from Shang et al. [102], and (c) Fostiropoulos et al. [28].

ence of the fluid velocities on the temperature near the interface, which shows that modeling with the VOF gradient method allows capturing the effect of the flow on the interfacial thermal field. Fig. 2.16(b) shows results reported by Kumar and Das [59] on the temperature distribution in a growing bubble during film boiling; the results show that simulation properly accounts for the temperature condition at the interface with saturation value. The figure shows that simulations adopting the VOF gradient method

allow quantification of the heat transfer mechanisms near the interface.

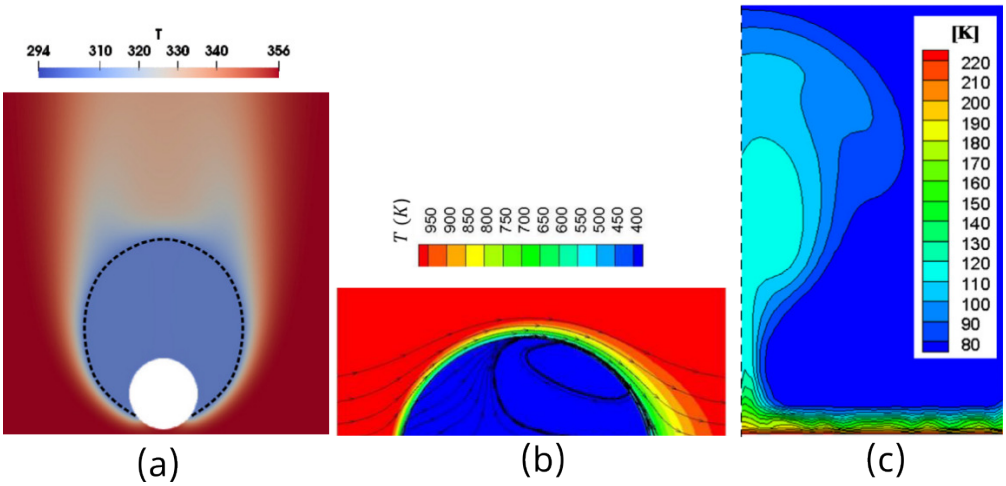

Figure 2.16: Temperature obtained by computer modeling adopting the VOF gradient method. Adapted from (a) Shang et al. [102], (b) Fostiropoulos et al. [28], and (c) Kumar and Das [59].

2.3.3. The Interface Reconstruction Method

Table 2.3 shows the publications applying the interface reconstruction method consisting of about 26% (7 publications) in the field of boiling modeling from 2018 to 2022. The interface reconstruction method uses Eq.(2.5) to compute the volumetric mass rate and estimates the interface surface area with a geometric interface reconstruction. Available works in the literature employing an interface reconstruction method do not provide a detailed description of the adopted algorithm. In addition, works that find the interface surface area with a geometric reconstruction lack a specific equation used to estimate the interface surface area.

Figure 2.17 compares simulated spherical bubble growth rates obtained with the inter-

Table 2.3: This table contains information on recently published pertinent research/research groups' articles that use the Interface Reconstruction method of calculation.

Articles Using the Interface Reconstruction Method			
Year	Author(s)	Simulation Details	Interfacial Thermal Behavior
2022	Zhao et al. [141]	2D-axisymmetric and 3D simulation of boiling, including leidenfrost	"Explicit thermal behavior near edge"
2022	Chang et al. [17]	2D simulation of film boiling in magnetic fields	Not contact line focused
2021	Guggilla et al. [35]	2D-axisymmetric, evaporating droplets on hot surface	Acceptable
2021	Franz et al. [29]	3D subcooled flow boiling at microgravity	Smeared across interface
2021	Bureš and Sato [14]	2D-axisymmetric evaporation and condensation	"Sharp interface temperatures given"
2020	Gholijani et al. [32]	3D drop impingement onto a hot wall	Not displayed
2018	Sato and Niceno [99]	3D simulation of pool boiling	Sharp interface modeling

face reconstruction method against theoretical results. In addition, results in Fig. 2.17(a) show that the interface reconstruction approach can generate simulations that are sensible to the grid cell size and with improved accuracy as the grid cell size becomes smaller. Fig. 2.17(b) shows that the simulation with the interface reconstruction approach can capture the theoretical results with a relative error of 1% or lower. The results indicate that the interface reconstruction leads to precise bubble growth rates, which implies a proper computation of heat and mass transfer mechanisms at the interface. [68]

Figure 2.18 shows temperature contours obtained with numerical simulations adopting

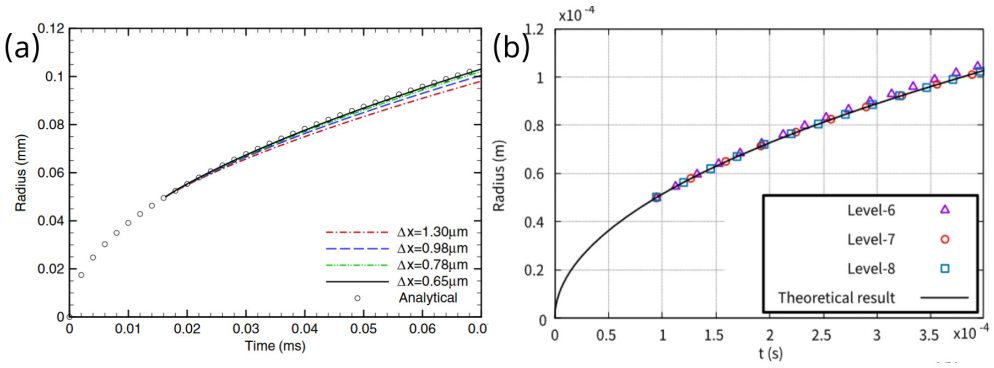

Figure 2.17: Spherical bubble growth rates obtained by computer modeling adopting the interface reconstruction method. Adapted from (a) Sato and Ničeno [96], and (b) Zhao et al. [141].

the interface reconstruction method. Fig. 2.18(b) shows temperature contours reported by Sato and Ničeno [96] in modeling of bubble growth over a heated surface. The temperature contour shows larger temperature gradients near the contact line and a drop in temperature to the saturation value at the interface. Zhao et al. [141] employed the interface reconstruction approach to simulate the falling of an evaporating droplet at gravity conditions. The model shows an interface with a saturation temperature influencing the surrounding fluid. The results in Figure 2.18 indicate that the simulations adopting the interface reconstruction approach captures the thermal interfacial conditions, which may contribute to clarifying the way heat transfer occurs near the interface.

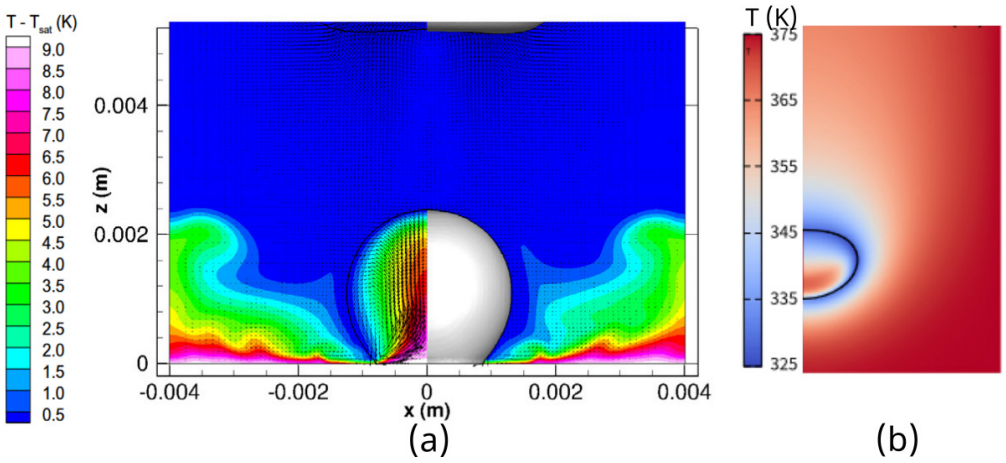

Figure 2.18: Temperature obtained by computer modeling adopting the interface reconstruction method. Adapted from (a) Sato and Ničeno [96], and (b) Zhao et al. [141].

2.3.4. Other Methods

This subsection offers an overview of those calculation techniques not given much attention in this document. Tables 2.4 and 2.5 are included for balance; though not exhaustive inclusively considering that human imagination, not any sort of strict practicality, is the limit of methods and techniques which may be presented in the course of the history of software. Of course, a degree of usefulness tends to be required for invention and/or publication, however it is much easier to quantify that which best fits experimental results than it is to offer a boundary on that which is reasonable.

Table 2.4 shows the publications applying methods other than these main three or not specifying how the interface was handled.

Table 2.5 references articles utilizing the Level-Set Method (LSM). The Level-set

Table 2.4: This table contains information on recently published pertinent research/research groups' articles outside of the three main categories of calculation previously detailed. This table being composed of "Non-specified or Atypical Calculation Methods," meaning the elements are an inclusive sample beyond the main three (Empirical Coefficients, VOF Gradient, and Interface Reconstruction)

Articles Using Non-specified or Atypical Calculation Methods

Year	Author(s)	Simulation Details	Method to Find Mass Transfer	Interfacial Thermal Behavior
2022	Onishi et al. [76]	2D modeling of a phase-change driven pump	Not accounted for	Not contact line focused
2022	Saha et al. [94]	1D, Flash boiling	Assumed spherical symmetry	No contours shown
2022	Huang et al. [43]	2D, crystal growth in solidification	Calculation of interface velocity	Not contact line focused
2021	Jafari [47]	1D phase change with deviations from equilibrium	Approximation with a square function of volume fraction and maximum interface area	Not contact line focused

method is interface-centric in a similar way that the VOF method is volume/mass-centric. This means that interface calculation is intrinsic to LSM. This sort of symmetry is deeper than these methods and is discussed in more detail in

Figure 2.17 compares simulated bubble growth rates obtained with the level-set method against theoretical and experimental results. Fig. 2.17(a) and (b) compare theoretical and simulation results for spherical bubble growth. Fig. 2.17(c) and (d) compare experimental and simulation results for bubble growth over a heated surface. Results in Fig. 2.17(a) and (b) show simulations with precise bubble growth rates. Also, the

Table 2.5: This table contains information on recently published pertinent research/research groups' articles that use the Level-set Method.

Articles Using the Level-set Method			
Year	Author(s)	Simulation Details	Interfacial Thermal Behavior
2022	Raut et al. [89]	2D-axisymmetric boiling at microgravity	Acceptable Contours
2019	Garg and Dhir [30]	3D simulation predicting boiling curve	Precise temperature near contact line
2019	Aktinol et al. [8]	2D axisymmetric boiling at microgravity	Apparent temperature near contact line

level-set approach is sensitive to the size of the computational grid, with improved accuracy as the grid gets refined and the existence of an optimal grid cell size. Results in Fig. 2.17(c) and (d) show good agreement between the simulated bubble growth over a heated surface and available experimental data. Results indicate that simulations with the level-set approach can reproduce experiments with an accuracy of 15%. These results imply that computer modeling can capture the mechanisms of heat and mass transfer at the interface in the simulation of nucleate boiling.

Figure 2.20 shows temperature contours obtained with numerical simulations adopting the level-set method in modeling bubble growth over a heated surface. The simulation in Fig. 2.20(a) assumed vapor to remain at a saturation temperature, whereas the simulation in Fig. 2.20(b) considered the heat transfer in the vapor phase. Results show the interface with a saturation temperature influencing the liquid and vapor temperature. The condition of saturation temperature at the interface forms a meniscus region with

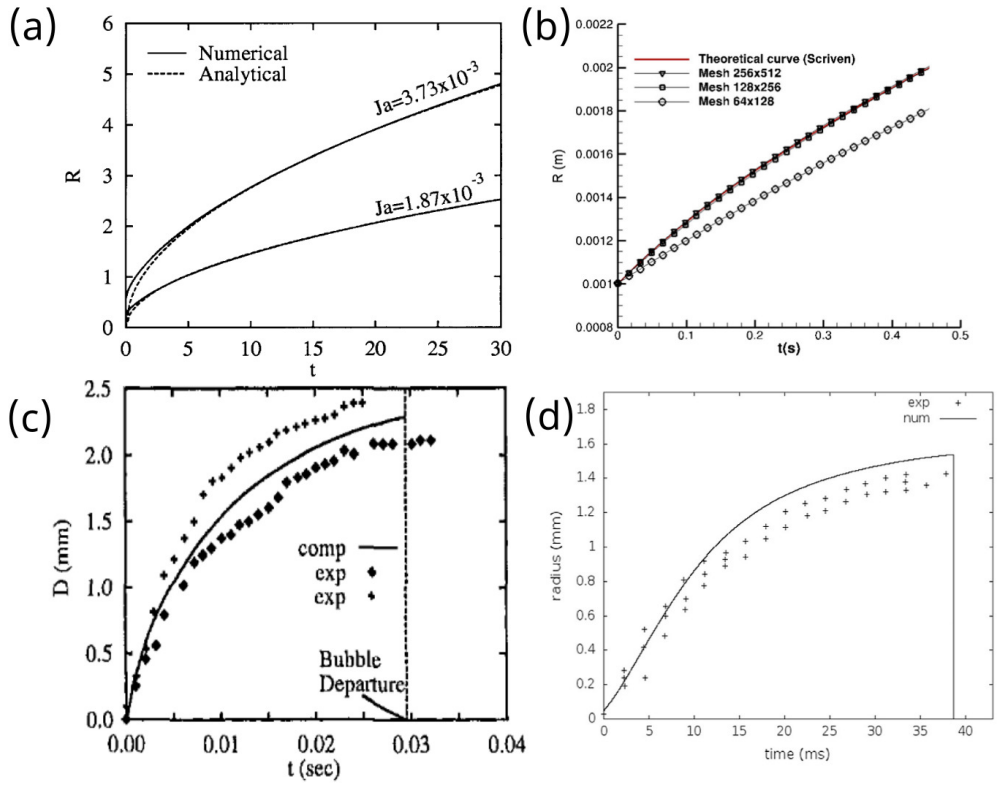

Figure 2.19: Spherical bubble growth rates obtained by computer modeling adopting the level-set method. Adapted from (a) Son [112], (b) Son et al. [111], (c) Tanguy et al. [118], and (d) Huber et al. [44]. Experimental data obtained by Siegel and Keshock [108].

large temperature gradients near the contact line. The large temperature gradients lead to a high convective heat transfer coefficient.

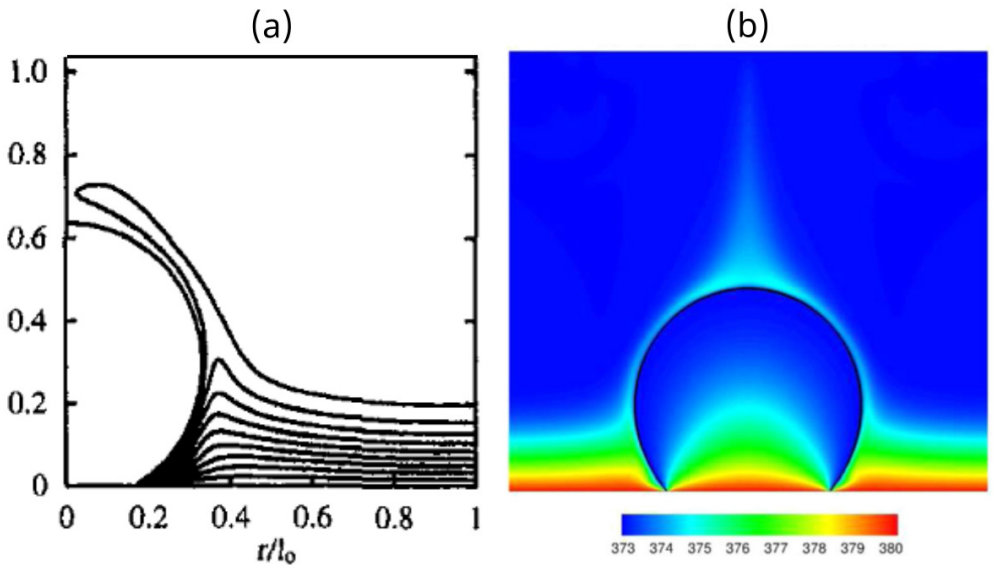

Figure 2.20: Temperature obtained by computer modeling adopting the level-set method. Adapted from (a) Son et al. [111], and (b) Huber et al. [44].

Chapter 3

Methods and Experimental

This part of the document covers the approach to the testing, Validation and Verification (V&V), and phenomenological problems the computational and simulation software has been applied. This includes the methods, parameters, and experiments (i.e. simulated cases).

Section 3.1.3 contains both a brief review of how the Piecewise Linear Interface Calculation - Analytic Size Based (PLIC-ASB) method functions and simultaneously within this section the integration of this keystone User Defined Function (UDF) into our larger simulation package.

3.1. The Proposed Piecewise Linear Interface Calculation - Analytic Size Based Method Defined

The following subsections provide a quick introduction into the formulation of this method, Section 3.1.1; contextual and description , Section 3.1.2; the details of implementation, Section 3.1.3; and application to example cases to aid in implementation and communication 3.1.4.

3.1.1. Formulation of the Piecewise Linear Interface Calculation - Analytic Size Based Method

Formulation:

From here the subsequent implementation is more digestible with a brief perspective on how the equations were developed.

When a Volume of Fluid (VOF) cell in a computational fluid dynamic (CFD) simulation has a combination of two phases (i.e. vapor and liquid) the cell is an interface cell. If measuring the volume fraction of one phase it would therefoe be fractional in these interface cells. For simulations where a single bifurcation describes the interface in these cells a calculation in each time step is to be made to determine the size of the interface. Considering a square cell as this interface cell it is from here that this section illustrates the calculation.

There are two (non-trivial) types of shapes that can be made when a square cell is divided into two sections by an intersecting line see Fig. 3.1. If the line cuts across adjacent sides a triangle is formed. If the line cuts across opposite sides a right angle trapezoid is formed (with a perfect rectangle a possibility that is considered the limit of this case not a third shape because of the mathematical formulation of these shapes).

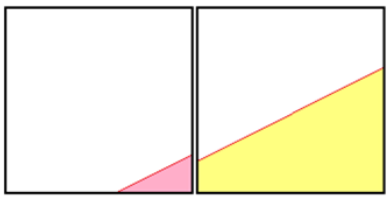

Figure 3.1: Illustration of the types of shapes possible from a linear interface in a square cell.

The quantity of interest at this point is the length of the line that is cutting across the square, l_{int}. The values that are known are the side lengths of the square, dx, the volume fraction of this cell, α_2, and the normal vector of this line is known due to a gradient of the surrounding volume fractions, shown in Fig. 3.2 as the blue arrow to referrence the prior rose trianlge and in Fig. 3.3 the yellow trapezoid.

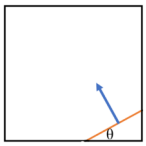

Figure 3.2: Illustration of the normal vector and yet unknown angle of interface for the rose triangle.

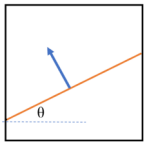

Figure 3.3: Illustration of the normal vector and yet unknown angle of interface for the yellow trapezoid.

Figure 3.4 shows a triangle constructed out of x and y components. If the angle was known the volume of any such triangle could be calculated as follows. Angle unknown, Problem 1!

$$\frac{xy}{2} = \frac{l_{int}^2}{2} \sin\theta \cos\theta = \alpha_2 dx^2 \tag{3.1}$$

Here α_2 is assumed to be equal or less than 0.5 ... otherwise $(1 - \alpha_2)dx^2)$ would have been used.

$0.5 > \alpha_2 > 0.5$ Problem 2!

Solving for l_{int}:

$$l_{int} = \left| \sqrt{\frac{2\ \alpha_2\ dx^2}{\sin\theta \cos\theta}} \right| \tag{3.2}$$

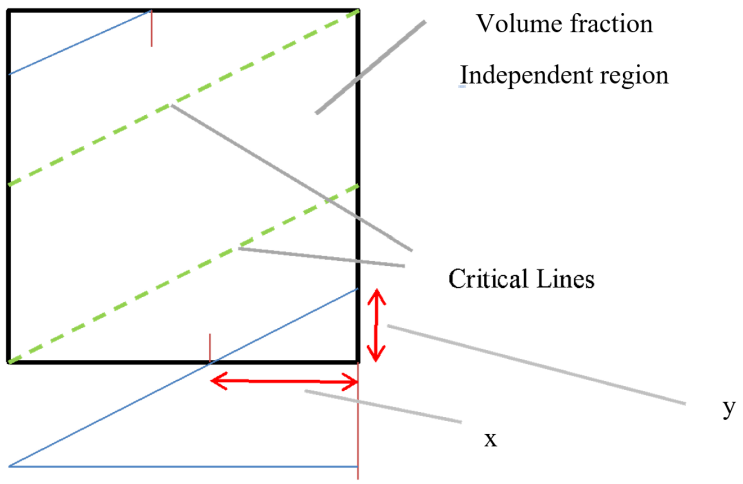

Figure 3.4: Illustration of cell geometry for PLIC-ASB solution. Here an arbitrary angle is chosen and the resulting criticle lines show the cross over from the volume fraction independent to the dependent equation.

For the right trapezoid case the calculation is even easier, except the matter of the angle is still just being assumed. If the top of bottom dashed line from Fig. 3.4 (or any line between them which will be of equal length) is l_{int} it follows that:

$$l_{int} \cos\theta = dx \tag{3.3}$$

Which yields:

$$l_{int} = \frac{dx}{\cos\theta} \tag{3.4}$$

Solving some of these problems:

Problem 1! Solved: How to find the angle from the normal vector. Thinking in terms of a simple method, an arbitrary orthogonal rotation could be applied to the normal vector to get a tangential vector. The components of this vector can be used to find the angle through the use of $\arctan$ but now a new problem- Problem 3! this could yield a high or low angle.

Problem 2! Solved: The volume fraction: The solutions are symmetric about 0.5, this will be addressed further below, so picking α_2 as either the high or the low value will stop the redundancy. So for $\alpha_2 > 0.5$, $(1 - \alpha_2)$ is the volume fraction for that case..

Problem 3! Solved: Just like the volume fraction a choice is made on the higher or lower angle for convention and then all of the problems can be funneled into that choice because (as will be address below, there is rotational invariance.

There remains a hidden problem which is converting this to an algorithm form for programming language or similar uses. Simply choosing which formula to use requires deciphering details from the input since after all it is relatively easy to calculate these values manually. The formulation results in the smaller length being the valid so that one may choose either formula, calculate and then compare against the other. This will be detailed further below.

Please continue to section 3.1.3 for more details on implementation.

3.1.2. Description of the Piecewise Linear Interface Calculation - Analytic Size Based Method

A brief context on the development of this method:

The Piecewise Linear Interface Calculation-Analytic Size Based method (PLIC-ASB) may be easily perceived as a sort of amalgamation of what has come before, especially given its hyphenated name, or a confluence of the best existing approaches, considering the background framing provided when discussing its merits. The reality of the PLIC-ASB is not just slightly different, it is a complete departure from this perception.

The PLIC-ASB is a novel approach to calculation where the influence of existing approaches was an influence only insofar as to inspire the innovation. The constraints used in its initial design were only that the algorithm would be constructed to rely on volume the fraction distribution for its input and have the most simple (i.e. least complex) final formulation. From this an analytical set of equations was determined to work as a suitable interface calculation where a so called "ansatz" would be used to make the final choice. Though "ansatz" was an imprecise hold over term from the formulator's (your author) former research experience, it was a useful mental check point. Upon further consideration, as the term was imprecise and easily replaced by an "If" statement both in and out of computer code, that is how the PLIC-ASB is expressed today, but the essence of the formulation has remained, and is otherwise unchanged

since its inception. For the most rigorous application there are methods to eliminate "If" statements readily available, but they are left out to avoid confusion.

The simplicity of the PLIC-ASB's formulation was only out matched by the absurdly simple techniques (e.g. setting interface cells to a constant value), but it is not the computational speed on its own that is the strength of this PLIC-ASB. Instead upon investigation with industry standards for the area of research application it was revealed that the accuracy and convergence, as any sort of interface calculation should have was in contrast to the recommended functions. The accuracy problem was due to an approximation in this existing method, the VOF gradient method, being treated as an exact solution. Apart from other researchers noting the problem with this method having sporadic behavior instead of converging to theoretical estimates with reduced mesh size the approximation issue is absent n the literature to our knowledge. Perhaps compounding this matter, the approximation is of a form that can seem as though it comes from a formal proof leading to an over reliance on these models.

In an effort to gain the best possible results from the VOF gradient method for a rigorous comparison it was found that the discrete gradient method could have an excessive impact on the final result. Such changes in the normal vector calculation method can lead to drastic changes in other methods, perhaps, but this is a compounding problem for the VOF gradient method. Best practices were utilized to give the VOF gradient a more than fair evaluation in the comparisons that will follow. The discrete gradient

method used for the PLIC-ASB method does not result in hypersensitivity, although, of course, some of the highly complicated discrete gradient calculations were not favored for use due to implementation and lack of improved normal vector calculation.

Systematically it seems there is a problem with the VOF gradient method, yet making a strong statement on this point is not the message of interest nor the research topic. It is of interest to see how the VOF gradient method can be best used and how the PLIC-ASB can continue to exceed these standards.

Testing methods, models, hypotheses, et cetera is a simple game when concerned to find if they perform on the task that we expect – how we expect. Unfortunately, it is not the things we know we are unaware of, but rather the unknowns that are hidden in a blind spot which itself is undetected.

The deficiency that has been found most acutely is that if a two-step discrete (digital) gradient is used instead of a single step gradient, the VOF gradient method has much higher rates of deviation from theoretically anticipated values for interface.

Although use of the single-step gradient improves the result, the underlying calculation is a multi-part approximation. If used with this knowledge and while understanding its limitations the VOF gradient method may be another reasonable method for continued use; however, the single step gradient implies both sides of the interface (at minimum) - must be calculated, not just the a single cell wide interest band in order to encapsulate

the total species/fraction change. Further errors may arise without bounding this and along with these it is also an added challenge to maintain sharp interfaces.

These results lead to testing the PLIC-ASB to reveal if there was any(or is as of) yet hidden flaw in the PLIC-ASB. This search for hidden flaw is expanded on in section 3.2.1, on verification for this User Defined Function (UDF).

3.1.3. Implementation of the Piecewise Linear Interface Calculation - Analytic Size Based Method

The implementation of the Piecewise Linear Interface Calculation-Analytic Size Based method (PLIC-ASB) is provided here in its succinct form. It is provided as a description of a method used in the described computational research and simulation although the method itself is part of the same. As such, a combined self-awareness and meta-level consideration is present by making the appropriate distinctions between the PLIC-ASB as a means versus an end as well as references to other parts of this document. Most generally, there is a light treatment of the means-to-an-end here, but otherwise in this section the PLIC-ASB is the ”means” whereas in the referenced section it is an ”end” unto itself.

These additional references provide a comprehensive description of the PLIC-ASB each with context and vantage so as to promote clarity amid the muddiness of this

intrinsic means/end challenge. For examples of pseudo-code see appendix A.1 which is useful for interested researchers of any type of software/CFD implementation or even for a streamlined view of the algorithmic thought to complement this part of the document. The use of this method in previously published works is also an ares for further reading. These include from this author, Shipkowski [105, 106] and its application with advisor, Perez-Raya [84, 85].

This section is divided into the following parts, stated here to keep a perspective into how the PLIC-ASB is utilized in the presented research (and how it may be applied to similar research).

1. **Start with simulation data in the proper format**
2. **Perform data transformation/translation**
3. **Apply governing equation**
4. **Continue/repeat** ...along with the cycle over all cells for other computational analysis/simulation steps being used...

These items are expanded upon individually below as:

1-Expanded (Start with simulation data in the proper format):

1.1 The Volume fraction of phase-2, α_2, for the interface cell

1.2 The components of the normal vector, $\hat{n}$, with respect to the interface:

n_x (x-component) and n_y (y-component)

This includes that an assumption of simulating on a type of square cell grid of computational cells is taking place. Adaptation to other simulations may be possible, but guessing at every combination is an absurd task. The described arrangement, a square mesh, may be: A. 2-dimensional, B. 3-dimensional cubic, or C. axial-symmetric via rotational symmetry.

- A. Is the most simple formulation, so it will be explained first.
- B. Has a number of different ways to handle so it will require additional treatment.
- C. will require the additional handling that is appropriate to this method of simulation.

In all of these cases, the important part is to understand the computational arrangement for a 2-dimensional perspective in order to extend the concepts.

2-Expanded (Perform data transformation/translation):

The input data is just three variables. From here, the data types/variables of the final governing equation are determined through the appropriate transformation/transla-

tions.

The volume fraction α_2 is transformed into $\alpha_{2,inv}$. This refers to a transformed value equivalent to the alternate phase or species, α_1, if that is the one with the smaller volume fraction, dx is the computational cell size, l_{int} is the 2-dimensional linear equivalent length-of-interface (with numbered indices 1 and 2 for tracking the volume fraction dependent and independent forms respectively[1]), $\hat{n}$ is the interface normal with components n_x and n_y, θ is the angle of inclination of the interface with respect to the x-axis after other rotations, or the angle with respect to the cell wall most close to parallel with the projected linear interface, and the subscript indices 1, 2 are place holders for distinction. With these understood equations (3.9) and (3.10) will provide a calculation of the interface on a cell-by cell basis over an algorithmic or computational process.

In order to have an analytic solution, a final calculation invariant symmetric degeneracy was exploited to reduce the level of unknowns. The flow chart (see Fig. 3.5) offers a visualization of the relative simplicity to use either algorithmic or analytical methods to implement the PLIC-ASB. The symmetric degeneracies are shown as converting the variables into alternate non-degenerate forms. The final step of solving is performed by making an assumption to avoid excessive numerical computational expense (note the last decision branch in the flow chart). The assumption is easily tested in computational

[1]There are two formulations which will be given depending on if the interface size is volumefraction dependent or not.

terms and may be expressed symbolically, meaning the formula remains analytic. The symmetry arguments for the PLIC-ASB are as follows:

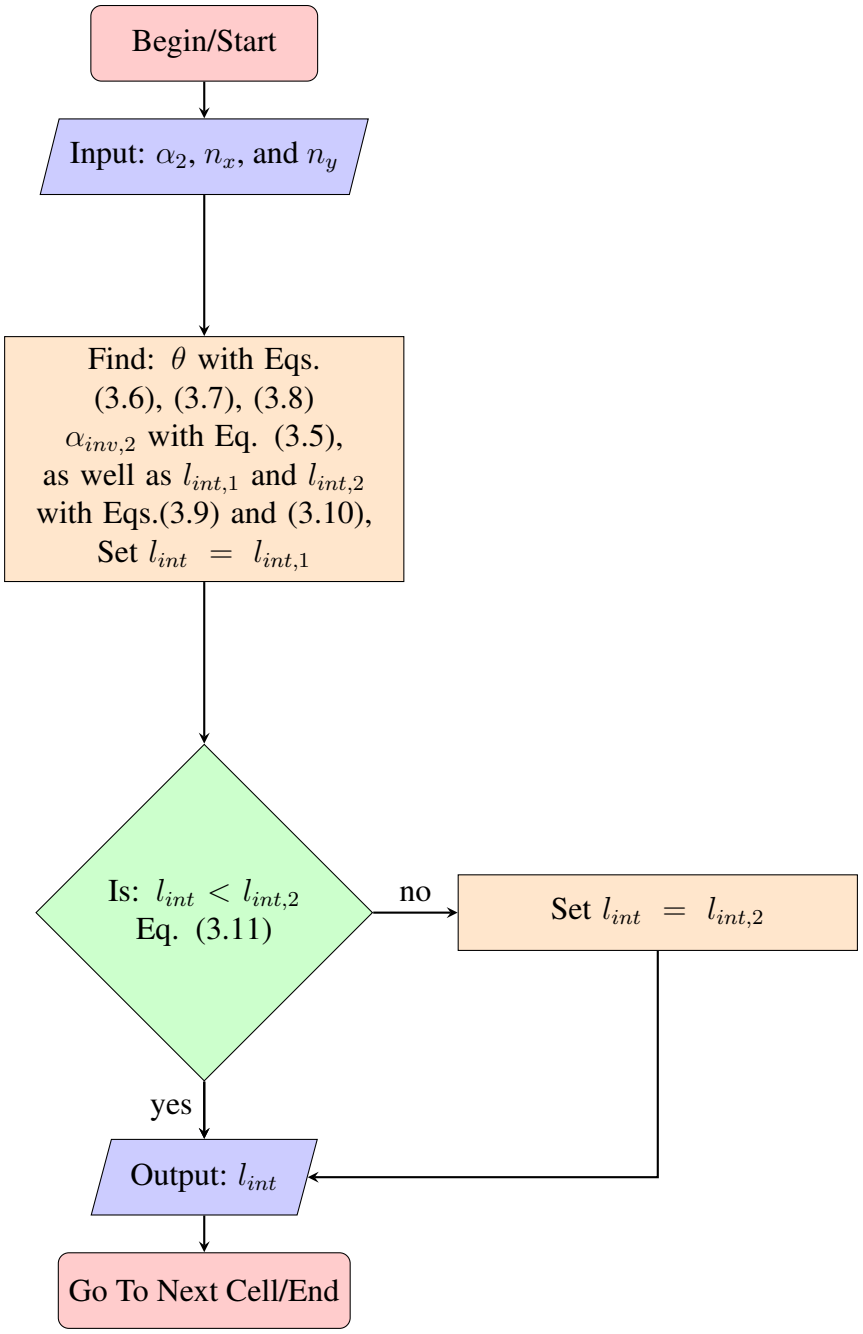

Figure 3.5: This is a simplified flow chart to visualize the PLIC-ASB as an algorithm. The trapezoids show input/output, the rectangles depict calculating processes, and the diamonds depict decisions. The rounded corner shapes remind that this calculation can be used as an instance or a part of a reoccurring loop.

Piecewise:

This is just to state that the interfaces do not have any continuity condition enforced at the boundaries of the cells. If an actual physical interface was reduced down to the set of linear areas or lines used for this size calculation that would be unphysical. In a similar way the simulation is not using any arbitrarily piecewise discontinuous interface. Instead, each succesive cell has an adjacent cell with an interface value. If there is a departure from this interdigitation or tearing would result from a complete discontinuity. The permitted level of discontinuity is only that no continuity function is imposed at cell boundaries. Radical discontinuity can occur if permitted, but this would be an improper use case for a physical simulation.

The importance of permitting a piecewise type fit is to get the closet linear representation. On the cell-to-cell level; this calculation only impacts transfer rates and physical size from one time step to the next at most (i.e. transfer rates may be made contingent by the user). Imposing continuity requires imposing at least first order curvature tracking.

Volume Fraction Transformation:

In order to reduce the degeneracy relating to the volume fraction the following is

applied. The transformation is contingent on the value being greater than 1/2. No change/transformation or species change of α_2 is needed such that $\alpha_{2,inv}$ = α_2 if it is equal to or less than 0.5, else, $\alpha_{2,inv}$ takes on the value of the other phase or species, that is $1-$ α = $\alpha_{2,inv}$ = α_1. This maintaining of the volume fraction at the midpoint of one half or less does not affect the length of the interface; it simply reduces computational complexity due to all of the calculations that can be done from the volume fraction midpoint or greater are a mirror reflection of those from the midpoint and less if the difference from the one half fraction is equivalent. It is important to note here that the lesser value of volume fraction used here is arbitrary and either the greater or the lesser side could have been used. A similar technique will be used to choose the small angle of the interface as opposed to the larger one, again just for convention in solving. However, if both were permitted at the same time and the data remained as it were prior to this simple transformation it would add complexity to be addressed later. In computation it could be made into a logical if statement potentially however that would add to the computational time.

$$\alpha_{2,inv} = \begin{cases} \alpha_2, & \text{if} \quad \alpha_2 < 0.5 \\ 1 - \alpha_2 = \alpha_1, & \text{if} \quad \alpha_2 \geq 0.5 \end{cases} \tag{3.5}$$

Determining the last part of the data transformation also a crucial component in un-

derstanding how the methodology of the PLIC-ASB relates to simple concepts from established geometric forms. Consider how if at any given time step during the course of a simulation (of the type described / or relating field where this method is used as a reconstruction algorithm) a simulation, the equation of the line traced by the interface on a 2-dimensional cell is unknown; even unknowable in many cases, but this is beyond the scope of this section. This unknown interface is approximated as a straight line in all methods of the PLIC type by definition. The reductive schema employed not so much in reducing to a linear fit- but in deciding the parameterization and weighting to these processes is what separates one PLIC formulation from another. Here an exact method (in contrast to an approximation or numerical method) with a maintained volume fraction and normal vector describe the enforced parameters. With the normal vector constraining the angle (i.e. slope) and the volume fraction constraining the position (i.e. intercept) an equation for a line can be solved. With the data in its current form this requires determining the slope (fundamentally). Any vector parallel to this linear interface will have the same slope. Therefore determining the correct angle can be done by rotating the interface normal because the normal vector is defined as perpendicular to the interface, therefore any arbitrary rotation of $\frac{\pi}{2}$ will enforce a parallel vector, which provides the desired line. This line, if used with the volume fraction information makes a section of the square cell such the x-vector and y-vector components reveal the possible angles of interface, the parts of the square cell above and below the line add up to 1 square in total. However the components of this line

can yield a number of angles depending on orientation. Per the rotational invariance paragraph, to follow, the interface size will not change under the rotation of the entire cell, therefore the cell is considered rotated into a position where the normal is mostly pointing in the y/up direction and secondarily in the x/right direction a simple function.

$$\overrightarrow{t} = R\,\hat{n} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \begin{pmatrix} n_x \\ n_y \end{pmatrix} = \begin{pmatrix} -n_y \\ n_x \end{pmatrix} \tag{3.6}$$

A point to take away from here is that the changing of the indicies and including a negative sign converts: $n_x, n_y \; into \; t_x, t_y$

$$\theta = \arctan\left(\left|\left(\frac{t_x}{t_y}\right)^{S_t(t_x,t_y)}\right|\right) \tag{3.7}$$

$$S_t(t_x, t_y) = \begin{cases} 1 & \text{if } \; t_x^2 < t_y^2 \\ -1 & \text{if } \; t_x^2 \geq t_y^2 \end{cases} \tag{3.8}$$

Rotational Invariance:

The actual true cardinal vector direction of the interface should have no bearing on its size, provided the relative values of the components of the interface parallel line are preserved. That is, a cell with a normal pointed in the y-direction rotated such that now it is pointed in the x-direction should in no way be expected to change in size. This is also true under reflection; if the x direction is replaced for the negative x direction for example. Rotation invariance is used to so that the angle θ is the smallest possible angle. Just like with the volume fraction argument, this is an arbitrary choice. The largest angle could just as easily have been chosen, but leaving it to any angle would have meant solving for all possibilities on all cells, except for the trivial case exceptions. The careful reader might notice that a fully x or y directed interface would be one such trivial case; the above examples were not given due to the rigour in mathematical detail, but rather their ease of communication. More detail is given in the parts of this document as described at the beginning of this section for that type of more rigorous consideration.

3-Expanded (Apply Governing Equation):

The next step is to apply the governing equation. Note that the referenced equations use dx as the computational cell size and l_{int} as the 2-dimensional linear equivalent length-of-interface with numbered indices for tracking (i.e. $l_{int,1}$ and $l_{int,1}$ are the volume fraction dependent and independent forms respectively). With the data given the

equations referenced are just listed out into the volume fraction dependent form: (3.9), the volume fraction independent form: (3.10), and the final determination: (3.11). These can be combined into a single form, but separated into these provides a more intuitive use prior to implementation in a computational package. The final determination is there so that the calculation can blindly solve the root, but it is required to be minimized; or more directly it is not physical for the linear interface length to exceed the independent formulation.

$$l_{int,1} = \left| \sqrt{\frac{2\,\alpha_{2,inv}(\mathrm{d}x)^2}{\cos\theta\ \sin\theta}} \right| \tag{3.9}$$

$$l_{int,2} = \frac{\mathrm{d}x}{\cos\theta} \tag{3.10}$$

$$l_{int} = \begin{cases} l_{int,1} \ \text{ if } \ l_{int,1} < l_{int,2} \\ \\ l_{int,2} \ \text{ if } \ l_{int,1} \geq l_{int,2} \end{cases} \tag{3.11}$$

4-Expanded (Continue/repeat): ...

This step is useful in making explicit assumptions common to the CFD field of re-

search. To this end, the following applies provided the calculation is to be done on all interface cells as is standard for the simulations conducted by the author and co-researchers as well as the vast majority of external simulations of this type. The alternative might be an ad hoc calculation, for example. This repeat-step is listed to make explicit that the PLIC-ASB is a process that is looped over all applicable interface cells. It should likewise be explicitly noted that its existence within said loop (i.e. a computational while loop) and repeated calling over a series of timesteps is not an iterative process towards the final calculation. Iterative processes of this type are those that are being solved numerically and in each iterative step moving closer to the solution value. Of course, some methods deviate from this and do not strictly adhere to this idealized description.

With the concept stated, that the PLIC-ASB is to be looped over all interface cells it is simple to imagine a successive repeating process nested as:

(**1, 2, 3, 4** then (**4-I, 4-II, 4-III, 4-IV** then (etc...)))

The nested layers are not needed outside of their in text ambiguity removal and pedagogical function. Object oriented programming languages provide an easy path for repeating even series of calls without nesting.

If restarting the process with:

4-I (Simulation data in the proper format):

α_2, n_x, and n_y the volume fraction of the interface cell and the normal vector components, respectively.

4-II (Perform data transformation/translation):

$\alpha_{2,inv}$ from α_2 using equation: (3.5)

θ from n_x and n_y using equations: (3.6), (3.8), (3.7)

4-III (Apply governing equation):

l_{int} from above using equations: (3.9),(3.10), (3.11)

4-IV (Continue/repeat): ...all cells...

3.1.4. Application of PLIC-ASB to eight example interface cells

This section illustrates the application of PLIC-ASB to eight different computational cells with an interface. The examples give the steps and interface transformation expected to occur when applying the developed method. Fig. 3.6 shows the eight cases consisting of interfaces with different volume fractions α_2 (here depicted as the non-shaded region) and inclinations given by the normal and tangential vectors. The tangential vectors may orient arbitrarily, as the user can choose any orthogonal rotation. Also it is important to be cautious that some CFD software packages will not return normalized or even reasonably bounded values for the normal vector, so do not rely on

these values to be the unit vectors without investigation.

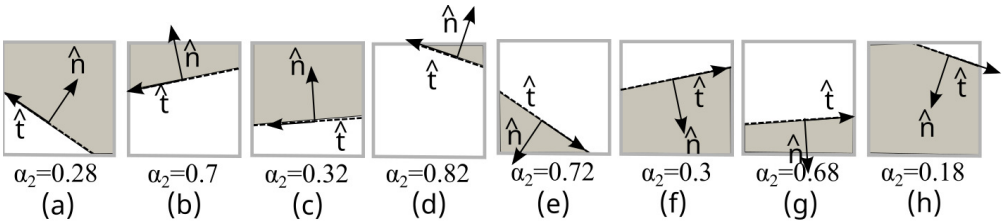

Figure 3.6: Eight computational cells with an interface. PLIC-ASB is applied to each of these cases for illustrative purposes. Adapted from prior work [106].

As illustrated in the flow diagram (see Fig. 3.5), the first step is finding the smallest angle θ formed between the interface and the x-axis to identify the interface inclination. Steps 1 and 2 below indicate the steps to find θ. Figure 3.7 shows (top section of the figure) the estimated lower angle between the interface and the x-axis.

- Step-1. Calculate the tangential vector with Eq. (3.6).
- Step-2. Utilize the tangential vectors to find the angle θ with Eq. (3.7).

After identifying the angle of inclination θ, PLIC-ASB reflects and inverts the interface if the volume fraction α_2 is greater than 0.5. Reflection and inversion are needed to generate the analytical equation. The following step is conducted to reflect and invert the interface. Figure 3.7 shows (i) the original interface, (ii) the reflected interface, and (iii) the inverted interface for the cases with a volume fraction α_2 greater than 0.5.

- Step-3. Reflect the interface to create a geometrical triangle that will be used to

generate the analytical equation. Utilize Eq. (3.5).

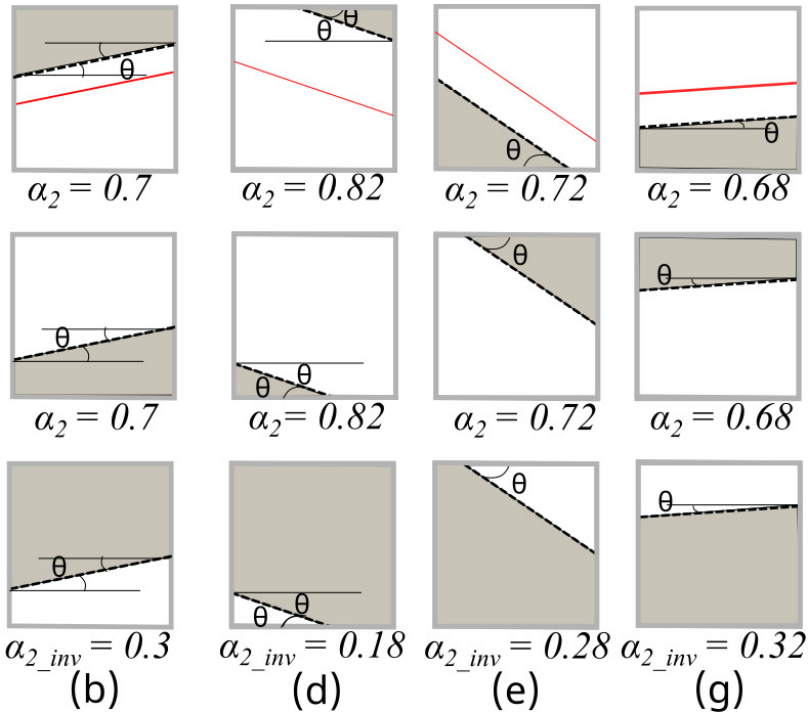

Figure 3.7: PLIC-ASB transformation of interfaces that are part of a cell with volume fraction greater than 0.5. Top: non-modified cells with the interface, middle: transformed cell, bottom: inverted cell. Adapted from prior work [106].

Figure 3.8 shows the analyzed cases after passing the reflection and inversion process. The geometrical triangle is now formed. Cases (a), (d), (e), and (h) created a triangle formed by phase-2 (white color phase) with the interface intersecting the edges of the computational cell. The area of the triangle is given by $A = l_{int}\cos\theta\ l_{int}\sin\theta/2$ or half the base times height. These cases are the volume fraction dependent cases.

The cases where small changes to the volume fraction does not change the interface size are the volume fraction independent cases. In cases (b), (c), (f), and (g) the area

solving potion can be neglected. The interface length is given by $l_{int} = dx/\cos\theta$. The prior result now in more detail.

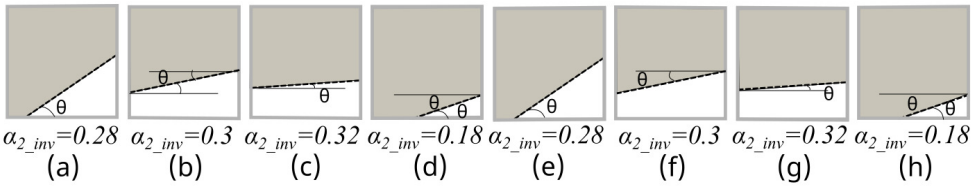

Figure 3.8: PLIC-ASB position of interface and phases of analyzed cases after the reflection and inversion process.

Next, PLIC-ASB utilizes the geometrical relations to derive algebraic equations that calculate the two possible values of the interface length. The following steps are performed.

- Step-4. Calculate $l_{int,1}$ with Eq. (3.9).
- Step-5. Calculate $l_{int,2}$ with Eq. (3.10).

Finally, PLIC-ASB applies step-6 to compare the calculated lengths and identify the size of the interface l_{int} in the computational cell.

- Step-6. Identify the interface size l_{int} by comparing $l_{int,1}$ and $l_{int,2}$ with Eq. (3.11).

In 2D flows, the interface length equals l_{int}. In 2D axisymmetric flows, the interface length should get multiplied by $2\pi r_c$ (where r_c is the distance from the axis of rotation

to the interface center) to get the interface surface area.

3.2. METHODS APPLIED TO SHOW THE ACCURACY OF THE DEVELOPED APPROACH

3.2.1. Verification and Validation

The Verification and Validation (V&V) of the PLIC-ASB here is in reference to an accommodated meaning in line with the computational science usage, distinct from other fields of software. It is Roache [90] who is responsible for the loosely quoted: Verification means you are solving the equations right. Validation means that you are solving the right equations. Though as Oberkampf, Trucano, and Hirsch [75] not the most technically accurate meaning of this term is for an entire model - not a single calculation or function. Yet the the scaled back activities certainly could apply to computation where an entire model is not in question. The use of descriptive ideal, possible, and ineffective option section helps this to be a particularly useful resource.

The Verification used here for the PLIC-ASB is about the action of this method in terms of what does the input and output look like and how unusual can it be made. It is to ensure that first, the User Defined Function (UDF) of the method is operating as it should be (insofar as user error can be discovered) and that 2. there are no functioning gaps from any proper operation no matter how unusual (exhaustive input check for error in the program).

Validation as it is used here is the process of ensuring that the PLIC-ASB method's results are not just precise, but accurate in the sense that what is being produced has significance beyond an approximation or worse an animation. An animation can make any sized bubble and do so with flawless precision. It is the realm of CFD, and any physics-based simulation to encounter new geometry, altered variables, or a previously unreachable /unstudied position with sensors and reveal at least a reasonable approximation of the underlying physics.

Verification through multiple implementations:

This PLIC-ASB method has been tested in the primary software, Ansys Fluent, but it was first tested in MS Excel. It has been operated in MATLAB, Mathematica, and a graphing calculator. The point of this is to explore the user capacity and trouble spots as well as have maximal access to every way to further explore the second half of verification.

After working out implementation and trialing for user error- verification calls for an exhaustive trial for program fault type sampling. There are limitations to a PLIC-type method, but that is not what is intended, rather the question is if the input is the exact wrong combination can a catastrophic failure of calculation occur?

It is also worth pointing out that reducing the cell size (i.e. mesh refinement) is nearly superfluous for this UDF. That is this is very context specific. If a shape has reduced

cell size it has the impact of reducing curvature and eliminating features which can be done through other means thereby permitting mesh position shifting and curvature (or feature) size to be de-coupled. The refining of mesh and the previously mentioned items are tested for individually and there will be discussed mesh refinement in the results section when the PLIC-ASB is not the only and crucial component at work in an overall multiphase model. If instead the purpose was to create a model attached to a particular software such as Fluent, where the PLIC-ASB was incorporated, then mesh refinement would be tied to verification and more over it would be almost pointless to proceed with any validation without said verification. This can seem to be a bit of a semantic point, yet the importance of such semantics and nuance was a key feature in Roache's direction on the subject [91] finding that getting tied up in sematntics is not the point but giving these important items words can alter the course of a project.

The unknown type flaws have been actively pursued after the basic input output testing. The closest version of this that has been found, was discovered proximal to when exhaustive testing was started. The problem input is a combination of very low angle and very low volume fractions. In these cases, it is possible for a cell's interface to be calculated improperly. What happens is that the low angle, in practice should be calculated as so close to horizontal that the cell interface is intersecting opposite cell sides. The error occurs when the low volume fraction places the interface close enough to the adjacent side that very exaggerated readings can occur. There is sensitivity testing

with the integration routine, shape rotation, and off integer alignment all in the static shape portion of the results that in part included determining if this was a significant concern for for similar but less drastic feature that may present in simulation.

Accuracy and Validation:

In the Results portion of this document, a number of tables containing data on the relative error of listed cases are provided. In these tables, the VOF gradient and the PLIC-ASB methods are compared. Gathering the entirety of this data it is evident that the PLIC-ASB does reduce the error of the VOF gradient method in contrast by well over half meeting the hypothesis statement. Averaging the data sets yielded a value of 28% where 100% would mean no change and 50% would mean half of the level of error present of the existing error. Most individual tables have values of 26%. Although the variance of the VOF calculated data is high, this has not led to significant outliers disrupting the data. There are 3 data points out of 34 that do not seem to reduce the error by half or more.

Validation has also been performed implicitly as this method is both theoretical (permiting its errors to be diretly calculated) and it was brought into use constantly trialed against other methods as part of the prior research phase development.

3.3. Experimental Setup

This Section, The Experimental Setup, refers to the key details in how the simulations or other computational tests were performed. In the following sections please find: **Data generation information 3.3.1**, the CFD software relies on a proper dataset for operation. This subsection describes the use of the integration routine and other methods for data generation, acquisition, and testing; **User Defined Functions (UDF) including Implementation of the PLIC-ASB 3.1.3**, provides a guide for the concept of UDFs and how to use the PLIC-ASB in any given researcher's code; **the technical details of the computational Fluid Dynamic (CFD) software setup 3.3.2** , provides the parameters/settings and software-specific methods needed to reproduce this research (along with the data) in lockstep; **and information and specifications about the associated computing resources used throughout this research 3.3.3**, provides information about the hardware and software (meaning software beyond Ansys-Fluent) that has gone into supporting this research with special interest in high-performance computing.

3.3.1. Data Generation: Integration Routine and CFD Tools

Simulations in the CFD software Ansys-Fluent require an initial dataset. Although there may be other ways of using the software, the use case of this software for the research discussed treats the data as essentially the initial conditions of the simulation.

The form of this data can be simple. For example a set of fractional values indicating the volume fraction of the corresponding cells at particular coordinates as governed by the organization of the set. Succinctly, the data is in the form of a vector container that has a proprietary transform into a an appropriate higher order tensor, usually an array. These data can have set extrinsic parameters such as temperature by phase or with a corresponding vector, and likewise intrinsic properties of each phase may be set such as density.

To begin the simulation every computational cell in the simulation domain will have an assigned value. Often a curve as defined by $f(x)$ or multiple such functions are used. The space bounded by the function(s) could be phase-2 while the space outside is phase-1. Any cells that the function intersects will take on an appropriate fractional value. Fluent includes tools to create these data, however it has been found that there is a level of approximation that is noticeable enough that it may influence believed static case convergence in the literature. If the same or similar methods are used to formulate the data and then read the data back, this should not be a surprise.

When creating a dataset, determining the appropriate shape is where the difficulty lies at a minimum. The larger difficulties include sourcing accurate data and sourcing said data when it is not/can not be functionally expressed or defined. For this reason and for testing reasons the integration routine was developed using primarily Wolfram Mathematica. Exact integration of simple and not-so-simple shapes piecewise over

individual regions (an imposed separated floating mesh) were compared with the data produced by Fluent after it was noticed that repeated output from fluent calling for an identical dataset resulted in minor deviations.

The integration routine operates to test in a separate way from any of the standard CFD packages and to generate reliable datasets. The data are produced by defining a mesh with the ability to adjust its total size, subdivision size (cell size), and number of divisions according to the needs of the job. Parallel to this mesh any shape or experiment of interest is then defined as the function f or a set of functions $f_0, f_1, ...$ as is expedient. It is often useful to use two or more functions to define a bounded region where one phase exists. Given that the mesh and function(s) to be evaluated are independent, a precise value of total volume, area, perimeter can be calculated if it is defined (i.e. Gabriel's Horn infinite area problem).

An additional operation of the integration routine is in converting image data to volume fraction data. This is the previously noted case where no function can be used to define the interface. The process used is to process the image into something like an intensity-based distribution. At its most simple this is use of gray-scale or black and white conversion options. Generally, there is image specific data that is favored which could help inform the discrimination. To give a simple example, consider an image with a good signal to noise ratio on blue and no noise on red but an incomplete signal. This image could be converted into two intensity distribution maps of these colors.

Any converted image does not need to have a one-to-one pixel-to-cell relation or even line up along the mesh. Although these are options and it is possible to even forgo the mesh and simply read off the converted data. At the same time, the mesh can translate a more detailed image into resultant cell values. The scenario where the image has fewer pixels than the mesh has cells is unremarkable and mostly used for a simulation that will require higher detail going forward.

3.3.2. Simulation settings

In this section, details are given on the particular setup for the Computational Fluid Dynamic (CFD) software based on the interest in testing benchmark type problems and evaluating the advantages of applying the PLIC-ASB method versus others. Of particular interest is finding any deficits in the PLIC-ASB in modeling multiphase flows where the interface surface area plays a primary role. Although somewhat arbitrary or a matter of convention, below is described phases of type 1 and 2. Liquid and vapor are more specific phases, states, or species. The interchange of these has only subtle meaning. In testing various phase-2 static and dynamic shapes immersed in a bulk of phase-1 with known interface surface areas (initial condition). Equation (3.12) is used to compute the volumetric mass transfer where the interface surface area comes from the VOF gradient method Eq.(3.13) or the PLIC-ASB method.

$$m'''_{lv} = m''_{lv} \frac{A_{int}}{V_{cell}} \tag{3.12}$$

$$A_{int} = |\nabla \alpha_v| V_{cell} \tag{3.13}$$

For the simulation, Eq.(3.14) is the VOF interface tracking equation used.

$$\rho_v \left(\frac{\partial}{\partial t} \alpha_v \ + \bigtriangledown \cdot \alpha_v \ \overrightarrow{v_v} \right) = m'''_{lv} \tag{3.14}$$

where ρ_v is the density of the vapor, α_v is the vapor-volume-fraction, v_2 is the velocity of the vapor phase, and m'''_{lv} is the volumetric mass rate liquid to vapor. The dependence of the volumetric mass rate with the interface surface area and volume of the cell is calculated by Eq.(3.12). Ansys-Fluent was used throughout the simulations detailed in this document. Both of the main approaches used in the simulations (VOF gradient and PLIC-ASB) require some modification of the software for Ansys-Fluent to compute the volumetric mass transfer. The mass flux is either constant (proportional) or it is calculated based on the temperature gradients normal to the interface, these are the adiabatic and non-adiabatic conditions, respectively.

$$m'' = -\frac{k_l}{h_{fg}} \frac{\partial T}{\partial n_{int}} \tag{3.15}$$

For the non-adiabatic conditions, Eq. (3.15) provides the relation of mass flux and temperature gradient, where k_l is the liquid thermal conductivity, h_{fg} is the latent heat of vaporization, T is the temperature, and n_{int} is the interface-normal directed variable. This variable is the quantification of the particular interface normal vector, generalized for commonly used Euclidean spaces $\mathbb{R}^2$ or $\mathbb{R}^3$ to the interface direction of arbitrary cells which can be obtuse put in general. However, for a spherical bubble centered on the coordinate system origin, this n_{int} becomes the variable r.

Naturalistic reasoning and the body of empirical evidence is used in concert with findings from Fedkiw, Gibou, and colleagues [27,33,34] who note how implementation of discrete temperatures can be made in error with an example using ice forming above the freezing point though misapplication. By this description the crucial aspect for this was also revealed. That is additional tunability can lead to unintended user error. Due to the need to include temperature dependence and the sharpest possible interfaces in the simulations we offer. Here the interface cells are set to a fixed temperature condition (cells where $\alpha_2 \neq 1$ or 0 , $T = T_{sat}$). Functional and perturbative tuning is possible, in case that helps to more closely represent fine detail from experiment, but it was not used.

Additional methods and algorithms are used in order to maintain a sharp interface for the simulations performed. Available techniques of interface tracking with mass transfer include, but are not limited to:

1. Smearing the interface into a restricted number of cells with an interface-sharpening equation [97, 98, 99] (this is in contrast with the more pervasive interface smearing technique which does not preserve a any sharp interface, however it remains a matter of mitigation).

2. The distribution of mass source terms in multiple cells around the interface [38, 61, 72].

3. The advection of the interfacial mass flux through the faces of the interface cells with a split approach [6, 36, 130, 131].

With the goal of simulating the sharpest interface a technique suited to the PLIC-ASB was considered. Perez-Raya and Kandlikar [84, 85] proposed a segregation algorithm that declares mass transfer only at interface-cells that share a face (i.e. cell boundary) with a vapor-cell; adding a second condition onto having a fractional composition. The method is compatible with the VOF gradient method, but designed for the PLIC-ASB. No smearing is required so the interface can remain as sharp as possible. Mass transfer is defined only at the select cells so that the simulation can avoid non-physical interface effects which usually require a diffuse interface to accommodate. It prevents

the advection of large mass fluxes to interface-cells that have a small interface.

The present work uses this segregation algorithm to simulate bubble growth, except in where alternative methods are noted or it is otherwise explicitly stated otherwise. This approach permits the sharpest possible interface for these simulations.

Technical specifications for the default CFD software used in the simulations includes: the continuity and momentum equations were solved with the SIMPLE (Semi-Implicit Method for Pressure Linked Equations) algorithm. The PRESTO (PREssure STaggering Option) scheme interpolated the pressure values at the faces in the momentum equation, see eq. (3.16) for the discretetized pressure equation, which is also used for momentum [126]. The Power-Law scheme discretized the convective terms in the energy equation. For the diffusive terms in the momentum and energy equations, a central difference second order discretization scheme was utilized. The VOF equation was solved with an explicit scheme of sharp interface modeling without interfacial anti-diffusion. The convergence criteria for the continuity, momentum, and energy equations was 110-7, unless otherwise specified.

In addition, Perez-Raya and Kandlikar give detailed information on these methods and the Ansys-Fluent customization [82, 84, 85].

For more details on the discretization schemes, the reader is directed to Ansys-Fluent documentation [110] and Versteeg and Malalasekera [126] as the description of the

details of the PRESTO! staggering scheme with associated equations requires the 6th chapter in the cited text to describe pages (179 -190). Figure 3.9 will guide the evaluation of the following indices i, I, j, J as to how they are related.

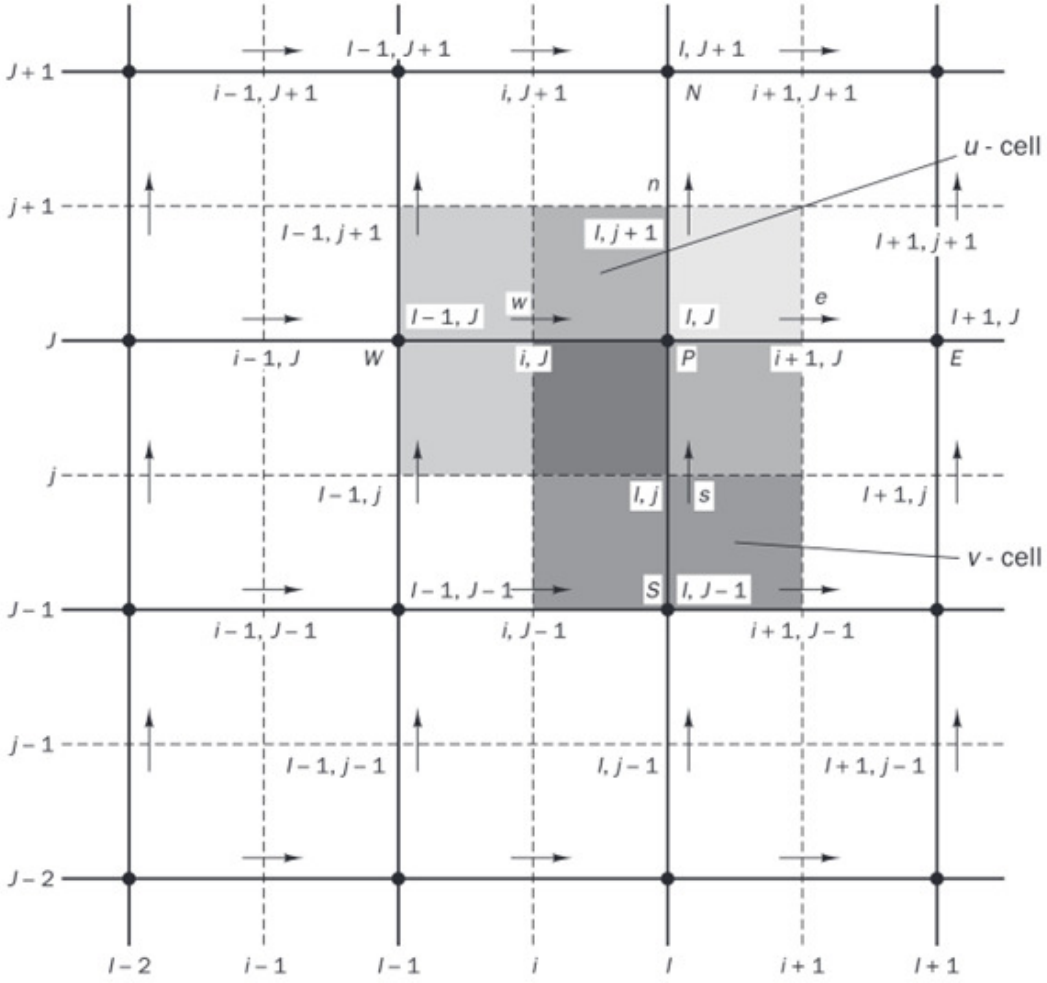

Figure 3.9: This diagram illustrated the meaning of the indices i, I, j, and J in the staggered scheme referrenced [126].

$$
\begin{aligned}
a_{I,J}p_{I,J} = & \\
& a_{I+1,J}p_{I+1,J} + a_{I-1,J}p_{I-1,J} + \\
& a_{I,J+1}p_{I,J+1} + a_{I,J-1}p_{I,J-1} + b_{I,J}
\end{aligned}
\tag{3.16}
$$

where

$$
a_{I,J} = a_{I+1,J} + a_{I-1,J} + a_{I,J+1} + a_{I,J-1} \tag{3.17}
$$

With coefficients where $Aisthecellface$:

$$
\begin{aligned}
a_{I+1,J} &= (\rho dA)_{i+1,J} \\
a_{I-1,J} &= (\rho dA)_{i,J} \\
a_{I,J+1} &= (\rho dA)_{I,j+1} \\
a_{I,J-1} &= (\rho dA)_{I,j} \\
b_{I,J} &= (\rho \hat{u} A)_{i,J} - (\rho \hat{u} A)_{i+1,J} + (\rho \hat{v} A)_{I,j} - (\rho \hat{v} A)_{I,j+1}
\end{aligned}
\tag{3.18}
$$

Where the pseudo velocity components $\hat{u}, \hat{v}$ are given as Eq. (3.19) and $d = A/a$ in respective index.

$$u_{i.J} = \hat{u}_{i,J} + d_{i,J}(p_{I-1,J} - p_{I,J})$$
$$v_{I.j} = \hat{v}_{I,j} + d_{I,J}(p_{I,J-1} - p_{I,J}) \tag{3.19}$$

3.3.3. Computing Resources

The combination of software and hardware used in the body of research described here is wide-reaching, from privately sourced to publicly available so much that it is likely excessive to scrupulously document minor contributors. The most notable for reasons of repeatability, audit, and disclosure are:

Hardware:

High performance computing cluster, RIT Research Computing Services

2304 cores (Intel Xeon Gold 6150 CPU at 2.70GHz)

24 TB RAM

100 Gbit/sec RoCEv2 interconnect (Mellanox MLX5/Juniper QFX210-64c)

(This helps to reveal why jobs that do not run in parallel will likely run fastest on a personal computer).

Personal computers (specifics listed where pertinent)

Software:

Simulation: Ansys-Fluent (19.2)

Wolfram Mathematica (mostly v11 and 12)

For additional details, example code, and script snippets, please see appendix (A), the computational appendix for this document.

Although not all of the simulations are suited for the use of the Research Computing Services (RC)/cluster computing, where applicable a number of simulations were performed using this resource. RC offers the ability to bring the power of many cpus on a problem which is valuable if the computational problem is able to be highly divided among a number of different cpus at the same time (i.e. suitable for parallel processing). Some programs are very easy to parallelize where as others are seemingly impossible. Thankfully the simulations from this research tend toward the former. Please

see [92] for added information/documentation on RC at RIT.

The RC SPORC (Scheduled Processing On Research Computing) at RIT is a High Performance Computing (HPC) cluster with Red Hat Enterprise-7 operating system which is familiar to those with HPC experience and to a degree Linux or bash/command line users. The S in SPORC is for scheduled which means that preparing jobs and submitting them in a mode for the scheduler to populate as appropriate (i.e. batch) is a great feature, despite almost no immediate user feedback from the system. The system provides a means to increase computing throughput and prevents users from waiting in a sort of queue themselves or planning use time around a restrictive schedule. Maximized processing is too nebulous to measure and even in the best cases would require some resources to triage input requests for allocation or allotment as well as operate the interface nodes for jobs to be submitted and upload down processes. Meta-analysis finds that not only is this the case, but that said allocation is a driving factor in quality of service (QoS), throughput, and other indicators despite raw processing speed for cluster, grid, or cloud-based HPC [45].

This should not be confused with meaning the processing power is of no consequence, rather that there are diminishing returns not a linear sort of cost-benefit increase. With that the software and hardware both are crucial for operations.

The use of HPCs for big data processing applications such as weather forecasting, astronomical, manufacturing optimization, commerce, energy exploration, speculative

markets, and other commercial interests contines to grow. A distributed multi-tier grid including international labs is the arrangement used for the large collaboration, CERN in handling high energy physics data processing and storage. Local clusters are more common to find in any given academic setting where particular job demand and funding may not lead to much HPC interest. Until recently the choice of such processing power for interested parties has been the HPC of access, however with continuing advancements such as on demand/pay for time and cloud infrastructure the capabilities of HPC are permitting more and more computationally demanding computational science to becoming a growing sub-discipline within fields previously unanticipated. For this reason we encourage development of this type of resource for the progression of computational science.

Optimization for the RC Cluster: From testing to improve the performance of these simulations it was discovered rather quickly that the RIT RC cluster issues real-time aka wall clock time not CPU time for processes in scheduler rotation. From this only the weakest link in resource management optimization could be performed. This is because job run time was primarily a function of the amount of resource demand, type of resource demand, and amount of on schedule rotation time. These items are mostly obscure to a user in a quantifiable measure. Despite this difficulty, this scheduling arrangement should not be considered problematic. The use case is what determines best practices and limits based on realtime are useful in both business settings and in

research areas where demand can be highly variable to permit accessibility. Further, there are of course methods to further test a program for best resource optimization, however in the cases stated here it was discovered that resources varied significantly enough between simulations so as to make such an investigation unfruitful for a user with some experience in high-performance computing and not a need to repeat very similar code. On the other side, if using a similar process over a steady stream of data this arrangement would be ideal.

Below is an overview of the workflow when using a resource such as the RIT RC cluster. It is omitted that one must open a terminal and log in generally the Secure SHell (SSH) protocol is needed and for this research MobaXterm was used. It should also be noted that one may exit the active session after submitting an executable to the scheduler without interrupting the anticipated completion of the job. It is good practice to monitor when an untested script/program is going to run for the first time and to check account activity for unanticipated behaviour such as inadvertent over-requesting resources or a job running with a typo leading to a null simulation.

The use of an HPC resource as a workflow is as follows:

1. Prepare an Ansys-Fluent program and data set
2. Prepare any User Defined Functions (UDF) for the program
3. Prepare a job-script for calling the Ansys-Fluent program, data, UDF(s) **The Executable**
4. Prepare an RC script calling the job to run per the scheduler, including loading the needed software/Run Time Environment (RTE)
5. Request a job through the script from item (4.)
6. Wait... Check that the job is running well, correct as needed.
7. If job/program completes in the allotted time, go to step (11.), else go to step (8.)
8. Use the last data output file as the data set in step (1.)
9. Update the script(s) in steps 3 (and 4 if needed) to correspond to the simulation time completed. (e.g. if 6 ms out of 10 ms completed change start time from 0 ms to 6 ms) More on this step: The general process is stitching together a complete simulation out of multiple sub-jobs. This step is included for cases where time limits do not permit the simulation to complete in one pass. Data near the end of pass 1 may be used as the initial data of pass 2
10. Go to item (4.)
11. Download the completed simulation data and stitch together if necessary

3.4. SIMULATION CASES

In this section a number of simulation cases are described. These cases are used for validation and verification (V&V), to uncover any areas of deficiency for the PLIC-ASB/surrounding methodology, to compare and contrast with established methods, and as an iterative path to improve the research team's capacity in the pursuit for phenomenological insight.

Figure (3.10) illustrates various static cases analyzed, including (with letters a, b, c, d referencing) those used in Fig (3.10):

(a) 2D centered circle: General test case

Nature: Time-dependent adiabatic, static

Formal: Off-integer cell alignment, various mesh sizes

(b) 2D and axisymmetric 3D quarter-spherical: Increased realism test case

Nature: Time-dependent non-adiabatic, Static

Formal: Off-integer cell alignment, various mesh sizes

(c) square rotated: special case used for V and V.

Static case used for added formal data

Shape rotated for cell-by-cell evaluation of corner and face cells at known angles

(d) semicircle-cut-out-square rotated: special case used for V and V

Static case used for added formal data

Shape rotated with the concept from (c) but added cell-by-cell analysis of well-defined curve

The circle radius is 0.1 mm, which corresponds to the radius of a small nucleated bubble.

Cases (a) and (b) are included to evaluate the effect of the grid cell size of 1, 5, and 10 μm with a bubble radius of 0.1 mm.

Cases(a) and (b) evaluate the effect of slight changes in the bubble radius with a grid cell

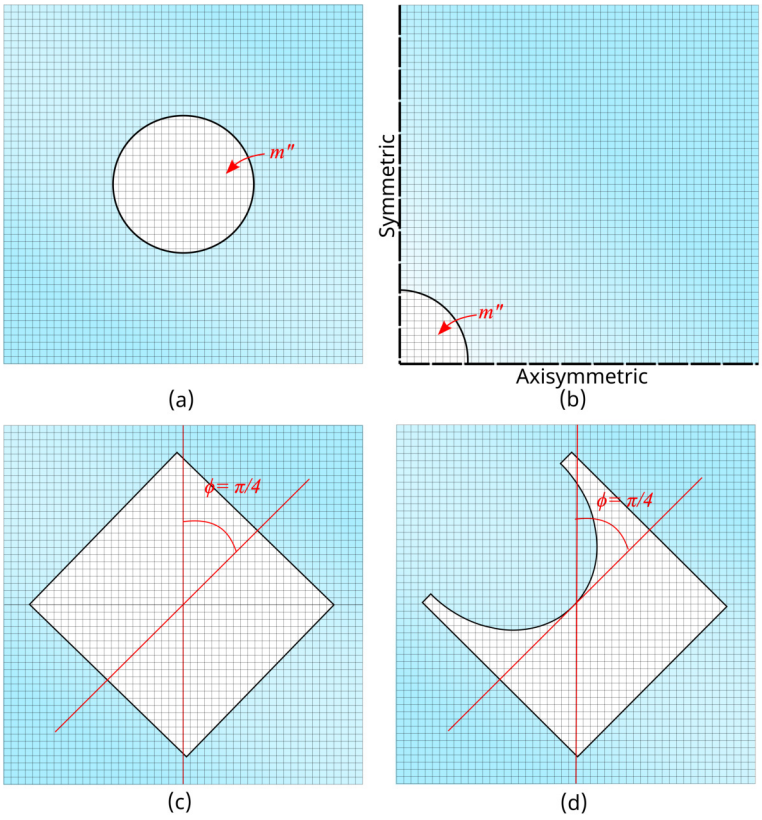

Figure 3.10: This figure depicts multiphase cases. (a) 2D centered circle, (b) 2D-axisymmetric quarter-spherical, (c) square rotated, (d) semicircle-cut-out-square rotated. Adapted from prior work [106].

size of 5 and 10 μm.

Case(c) evaluates the effect of the angle of rotation (0, 22.5, 45, 67.5, and 90°) in a square shape and a grid cell size of 1 μm.

Case(d) evaluates the accuracy in a circular and square geometry rotated at an angle of 45°.

Figures 3.10(a) and 3.10(b) illustrate the dynamic cases analyzed, including (a) 2D cen-

tered growing bubble and (b) 2D-axisymmetric quarter-bubble. Case(a) consists of bubble growth in adiabatic liquid with a constant mass flux. Case(b) consists of bubble growth in superheated liquid with a variable mass flux. The initial radius of the bubble was 0.1 mm, and the working fluid was water at 1 atmosphere with a 5 K superheated level (non-adiabatic bubble growth). Case(a) considered grid cell sizes of 10 and 1 μm. This leads to a relatively lower resolution simulation. It is not pushing the limit of what is possible for these methods, but by remaining in the range where every cell can be hand calculated this is suitable for noting differences and evaluating them at a causal level. Case(b) considered smaller grid cell sizes of 1, 0.6, and 0.2 μm to capture the thermal film near the bubble edge. In the simulation of case(b), the mass flux is estimated based on the heat flux at the interface [84].

In bubble growth with a constant mass flux (adiabatic bubble growth), the theoretical expression for the radius as a function of time is given by:

$$R(t) = R_0 + \frac{\dot{m}''}{\rho_v}\, t \tag{3.20}$$

For the non-adiabatic bubble growth cases, the mass flux calculation is dependent on the temperature gradient at the interface. The temperature gradient evolves in time due to changes/interactions in the interface-liquid system such as the development of the thermal layer, and the surrounding temperature distribution (i.e. liquid nearing saturation or

superheating depending on simulation). Scriven's theory on spherically symmetric phase growth [101], is derived with an infinite medium region to address the known physics and provide the general solution. It is simplified for growth controlled by heat and mass transport. The initial conditions consist of a small spherical bubble immersed in a liquid medium, the vapor and the interface have a temperature $T = T_{sat}$, and the liquid has a temperature $T_{(r,\, t=0)} = T_B$. As time evolves, the interaction between the liquid and the interface creates a thermal film in the liquid adjacent to the interface. The radial depth of this thermal film increases with the time evolution due to bubble-liquid conduction heat transfer.

From Scriven's solution [101], the bubble radius is Eq. (3.21)

$$R(t) = R_0 + 2\beta\sqrt{\alpha_l\, t} \tag{3.21}$$

where β is a growth constant obtained from solving Eq. (3.22).

$$2\beta^3 \left[h_{fg} + (T_B - T_{sat})\,(c_l - c_g)\right] \int_{\beta}^{\infty} \lambda^{-2} exp\left(-\lambda^2 - 2\varepsilon\beta^3\lambda^{-1}\right) d\lambda = \frac{c_l\,(T_B - T_{sat})\,\rho_l/\rho_g}{exp\,(\lambda^2 + 2\epsilon\beta^2)} \tag{3.22}$$

From mass conservation, Eq. (3.23) calculates the theoretical fluid velocity magnitude in the radial direction at a specific time as:

$$u_r = \frac{R^2}{r^2}\left(1 - \frac{\rho_v}{\rho_l}\right) R'(t) \tag{3.23}$$

where R is the interface location, r is the radial location, and $R'(t)$ is the interface velocity. The mean absolute error (MAE) estimated the error between the theoretical and the simulated velocities as:

$$MAE = \frac{1}{N}\sum_{i=1}^{N} |u_{r,th,i} - u_{r,sim,i}| \tag{3.24}$$

where N is the number of computational cells in the simulation domain, $u_{r,th,i}$ is the theoretical velocity magnitude at the cell i, and $u_{r,sim,i}$ is the velocity magnitude from the simulation at the cell i. Eq. (3.25) shows the formula that computes the relative error between the expected and the estimated interface surface area:

$$Rel.error = \frac{A_e - A_m}{A_e} \tag{3.25}$$

where A_e is the expected interface surface area or interface length, and A_m is the interface surface area calculated with the method (VOF gradient or PLIC-ASB). It is paramount to observe that a negative value of the relative error indicates an overestimation of the expected value.

3.4.1. Bubble Growth Over a Heated Plate

This next case for consideration is of interest because it represents an idealized realistic boiling experimental scenario. In pool boiling experiments a substrate is oriented below the developing bubble and is the source for continued growth. Some arrangements require special caution in monitoring for CHF as melting experimental components will occur directly after [67, 81]. In a pool boiling setup, after the Leidenfrost transition, only a reduction in heat transfer is the immediate risk which may include damage to any surface enhancements.

Here we extend testing of the PLIC-ASB to a common application where high resolution is critical. The degree of difference between the proposed research and the VOF gradient method will be quantified by comparison with available experimental data. Eventually, we will perform multiphase modeling with multiple bubbles occurring with accurate heat and mass transfer mechanisms that will help us predict boiling performance in large-scale systems.

Figure 3.11 shows the analyzed case of bubble growth over a heated surface. The initial conditions consider a small bubble of 0.1 mm in radius seated over the heated surface and forming a contact angle ϕ of 40°. Also, the initial conditions assume the liquid to be superheated with a linear temperature profile of 373.15 K at the surface and 379.35 K at a distance of 1 mm. Gravity conditions are taken into consideration

per experimental apparatus resulting in buoyancy forces that contribute to the bubble departure. In addition, the simulation considered microlayer evaporation to account for the evaporative conditions at the triple-phase contact line.

The microlayer model assumes 1D fluid flow along the horizontal direction and 1D heat transfer along the vertical direction. It also accounts for the effect of the molecular forces with a modified Young-Laplace equation, Eq. (3.31) [85, 127]. Under these conditions, the governing equations for fluid, heat, and mass transport are as follows.

Reference nomenclature for the following equations in order of appearance:

u Fluid velocity in ξ

η Axial coordinate

ξ Radial coordinate

μ Dynamic viscosity

P_l or P_v Pressure of the liquid or vapor, respectively

δ Film thickness

m''_e Mass flux (evaporation)

m''_{mic} Interfacial mass flux in the microlayer

ρ_v Vapor density

q''_e Interfacial heat flux in the microlayer (evaporation)

$T_{w,mic}$ Surface temperature at the microlayer

T_{sat} Saturation temperature

h_{fg} Latent heat of evaporation

R_{int} Interface resistance/simplifying notation

k_l Thermal conductivity of the liquid

f Evaporation coefficient

R Ideal gas constant

σ Surface tension

K Curvature

A Hamaker constant

Equations (3.26) and (3.27) give the velocity distribution in the microlayer and pressure-velocity relation. The standard no slip at the solid-liquid interface and the (vanishing) sheer stress in liquid-vapor interface is neglected.

$$u(\eta) = \frac{1}{\mu}\left(\frac{dP_l}{d\xi}\right)\left(\frac{\eta^2}{2} - \delta\eta\right) \tag{3.26}$$

$$\frac{\mathrm{d}P_l}{\mathrm{d}\xi} = \mu\frac{\mathrm{d}^2u}{\mathrm{d}\eta^2} \tag{3.27}$$

Next, Eqs. (3.28) and (3.29) from Wayner, Kao, and LaCroix's application of Clausius-Clapeyron equation [128], $q''_e = h_{fg}m''_e$, and Kamotani's 1D heat transfer reduction

for the microlayer [53]. Where R_{int} is given as an interfacial resistance term useful for streamlining the equation.

$$m_e{}'' = \frac{d(m_{mic}{}''\delta)}{d\xi} = -\frac{\rho_v}{3\mu}\frac{d}{d\xi}\left(\delta^3\frac{dP_l}{d\xi}\right) \tag{3.28}$$

$$q_e{}'' = \frac{(T_{w,mic} - T_{sat}) + \frac{T_{sat}}{h_{fg}\,\rho_l}(P_l - P_v)}{R_{int} + \delta/k_l} \tag{3.29}$$

$$R_{int} = \left(\frac{2-f}{f}\right)\frac{T_{sat}\sqrt{2\pi R\,T_{sat}}}{h_{fg}^2\;\rho_v} \tag{3.30}$$

$$P_l - P_v = -\sigma K - \frac{A}{\delta^3} \tag{3.31}$$

In Eq. (3.28), $m_{mic}{}''\delta$ gives the mass flux per unit contact line length; it assumes that mass flux in the microlayer is given by $m_{mic}{}'' = \rho\bar{u}$, where $\bar{u}$ is the mean liquid velocity. The interface curvature K in Eq. (3.31), the modified Young-Laplace equation, is given in Eq. (3.32)

$$K = \frac{\frac{\mathrm{d}^2\delta}{\mathrm{d}\xi^2}}{\left(1 + \left(\frac{\mathrm{d}\delta}{\mathrm{d}\xi}\right)^2\right)^{3/2}} \tag{3.32}$$

The solution to the system of differential equations gives the mass flux in the microlayer. To couple the microlayer model to the numerical simulation, the obtained magnitude of the microlayer evaporation is applied to the numerical simulation as a mass source term in the computational cells containing the triple-phase contact line. Perez-Raya and Kandlikar [85] give a detailed description of the methods followed to solve the microlayer model and its implementation in Ansys-Fluent.

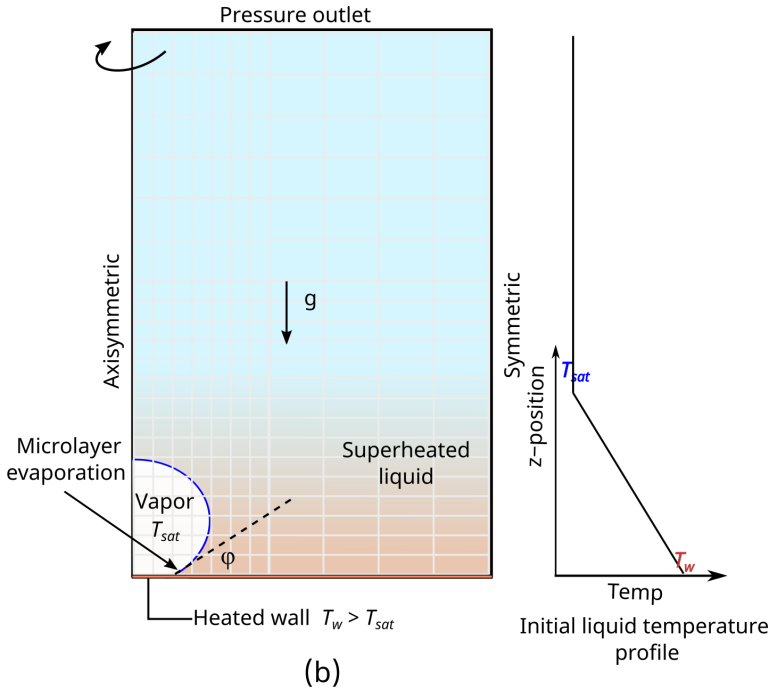

Figure 3.11: Schematic of a nucleate boiling simulation where the bubble is growing due to the condition of a heated surface. Adapted from prior work [83].

Chapter 4

Results

In this part of the document results are reported from simulations using the Piecewise Linear Interface Calculation-Analytic Size Based (PLIC-ASB) method. These results are compared with anticipated values found based on theoretical predictions and empirical findings. The quality of the PLIC-ASB results are evaluated on their own merits as well as against the VOF gradient method due to the similar research space these two methods occupy.

These items are placed in conformity with the objectives laid out in Section 1.1 for greater clarity. The findings from the results here confirm the remaining open points of the hypothesis. These items include from the starting point of test cases for further Verification and Validation (V & V) to realistic simulations in line with experimental data that has been considered and retested by previous researchers [108].

4.1. APPLYING VOF GRADIENT AND PLIC-ASB METHODS TO STATIC INTERFACES

The first item to report is a circle from the general configuration Section 3.3, Fig. (3.10) type (a). Based on the objectives of Section 1.1.2, this test is part of Objective 3.

Table 4.1 shows the quantified interface length obtained with a 2D circular and static multiphase shape. Fig. (3.10)(a) provides an illustration of the analyzed 2D circular shape; the radii of the circle are 0.1, 0.101, 0.105, and 0.107 mm, and the grid cell sizes are 1, 5, and 10 μm. The magnitude of the radius and the grid cell correspond to conventional values utilized in research related to multiphase modeling [24,98,118,141]. The interested reader can calculate the value as $2\ \pi\ r$. The mesh refinement did not show much effect for either method here. As is evident in the table the noise between trials of the same cell size has a greater impact on the PLIC-ASB method, and the VOF gradient method shows the reverse anticipated relationship. However, this is not enough data or a broad enough sample for any conclusions.

The other change shown in the table is the slight variations in the radius help to evaluate interfaces of various orientations in computational cells without respect for any accident of alignment due to numerical matching of cell size and feature size. Taking this further, the circle was also placed off-center to ensure no special lining up was occurring.

The table shows the expected and computed lengths and the quantified error for the various analyzed cases. The expected length ranges from 0.628 to 0.672 mm. Results revealed

Table 4.1: Comparison of VOF gradient and PLIC-ASB method in calculating interface length in 2D circular objects case (a) from Fig. 3.10

Data from Case (a)						
Category	Grid cell size (μm)	Expected (mm)	VOF gradient (mm)	PLIC-ASB (mm)	Error VOF gradient (%)	Error PLIC-ASB (%)
Centered (Ro=0.1 mm)	1	0.628	0.525	0.6505	16.436	-3.538
Centered (Ro=0.1 mm)	5	0.628	0.507	0.6403	19.258	-1.902
Centered (Ro=0.1 mm)	10	0.628	0.481	0.6172	23.469	1.773
Centered - Off Integer (Ro=0.101 mm)	10	0.635	0.541	0.6829	14.825	-7.607
Centered - Off Integer (Ro=0.105 mm)	10	0.66	0.571	0.688	13.48	-4.285
Centered - Off Integer (Ro=0.107 mm)	10	0.672	0.572	0.7045	14.975	-4.788

that the computed length with VOF gradient ranges from 0.481 to 0.572 mm whereas the computed length with PLIC-ASB ranges from 0.617 to 0.7045 mm. The computed error ranges from 13% to 23% with the VOF gradient method and 1.8% to 7.6% with the PLIC-ASB.

It is important to highlight that the VOF method average relative error is about 17%, whereas this error quantity with the proposed PLIC-ASB method is about 3%. These values, as well as all but one from the table, are well below the threshold from the hypothesis. The obtained results demonstrate that the proposed approach can more accurately compute the interface length.

The next item to report is an axisymmetric sphere from general configuration section 3.3, Fig. (3.10) type (b). Based on the objectives of section 1.1.2, this test is part of Objective 3.

Table 4.2 shows the results of quantifying the interface surface area within an axisymmetric multiphase simulation. Fig. (3.10)(b) depicts the configuration of the analyzed 2D-axisymmetric shape consisting of a hemispherical bubble. The radii are 0.0985, 0.0995, 0.1, 0.1015, and 0.1025 mm, and the grid cell sizes are 1, 5, and 10 μm. The interested reader can calculate the area as $4\pi r$. The table only shows 3 decimal places. The table shows the expected and computed areas and the quantified error for the various analyzed cases. The expected interface surface area ranges from 0.122 to 0.139 mm^2. Results show that the VOF gradient method computes the area with a minimum error of 10% and a maximum error of 17%, whereas the proposed PLIC-ASB computes the area with a minimum error of 0.1% and a maximum error of 11%. In addition, surface area computations have a mean-standard deviation of 14$\pm$3% with the VOF gradient method and -3$\pm$5% with the proposed PLIC-ASB method. These values as well as those from the table are well below the threshold from the hypothesis. The proposed PLIC-ASB can generate highly accurate interface surface area estimations, allowing more accurate multiphase flow modeling. Here the standard deviation is only greater than the error itself for the PLIC-ASB method.

The next item to report is the rotated test geometric shapes from the integration routine image reconstruction combined methodology. See the general configuration section 3.3, Fig. (3.10) types (c) and (d) for a depiction. Based on the objectives of section 1.1.2, this test is part of Objectives 2 and 3.

Table 4.3 shows the results of computing interface lengths in cases of square rotated

Table 4.2: Table comparing the VOF gradient and PLIC-ASB method in estimating the interface area of 2D-axisymmetric spherical bubbles, case (b) from Fig.3.10

Interface Area Data for the Bubble in Case (b)						
Category	Grid cell size (μm)	Expected Area (mm^2)	VOF gradient (mm^2)	PLIC-ASB (mm^2)	Error VOF gradient (%)	Error PLIC-ASB (%)
Corner (Ro=0.1 mm)	1	0.126	0.104	0.118	17.047	5.896
Corner (Ro=0.1 mm)	5	0.126	0.106	0.126	15.3	0.121
Corner (Ro=0.1 mm)	10	0.126	0.107	0.13	14.793	-3.247
Corner: Off- Integer (Ro=0.1005 mm)	1	0.127	0.107	0.133	15.639	-5.102
Corner: Off- Integer (Ro=0.1025 mm)	5	0.132	0.106	0.147	19.381	-11.167
Corner: Off- Integer (Ro=0.105 mm)	10	0.139	0.125	0.15	9.987	-7.996
Corner: Off- Integer (Ro=0.0985 mm)	5	0.122	0.107	0.124	11.829	-1.723
Corner: Off- Integer (Ro=0.0995 mm)	5	0.124	0.11	0.128	11.584	-3.107
Corner: Off- Integer (Ro=0.1015 mm)	5	0.129	0.113	0.136	12.636	-5.002
				Mean: Std Dev:	14 $\pm$ 3	-3 $\pm$ 5

multiphase shape Fig. (3.10)(c), and semicircle-cut-out-square rotated Fig. (3.10)(d). The analysis varied the rotation angle and the size of the multiphase shapes. The domain length was 250 unit cells, the square length is 151 cells, and the semicircle length is 70.5 unit cells. The angles of rotation are 0 (R 0), $0.25\pi/4$ (R 0.25), $0.5\pi/4$ (R 0.5), $0.75\pi/4$ (R 0.75), and $1\pi/4$ (R 1), and the grid cell size is 1 μm. The table shows the expected and computed lengths and the quantified error for the various analyzed cases. Theoretically and with Wolfram Mathematica, the data generator, the expected interface length ranges from 2.416 to 2.738 mm, calculated from the perimeter of a circle and a square. Results indicate that the computed length with VOF gradient ranges from 1.806 to 2.517 mm

whereas the computed length with PLIC-ASB ranges from 2.41 to 2.93 mm. Interface size computations have a mean-standard deviation of 13$\pm$8% with the VOF gradient method and -3$\pm$3% with the proposed PLIC-ASB method. In some instances, PLIC-ASB method shows extremely small errors of the order of 0.01%. The obtained results demonstrate that the proposed approach can more accurately compute the interface size. These values as well as those from the table are well below the threshold from the hypothesis. Here the standard deviation is only greater than the error itself for the PLIC-ASB method.

Further these data were designed specifically to evaluate known sources of error by introducing features such as the corners that are impossible to properly calculate as a linear interface. The PLIC-ASB including error which is unavoidable based on the method making these linear assumptions was what was the goal of this investigation in part. After careful analysis of the output for every cell here as well as data not reported for calibration that had suspiciously low error, it was found that: Yes, the impossible to represent cells (i.e. inclusive corners) were the source of the majority of the PLIC-ASB error contribution. Oddly removing these same cells from the VOF gradient method results did not impact the error in a drastic way. This was useful in that the source of error was gaining understanding for the PLIC-ASB method and a deeper insight into the limitations of the VOF gradient method was found.

Table 4.3: Table of data comparing the VOF gradient and PLIC-ASB methods in calculating the cummulative interface at different rotations of the geometric shape cases (c) and (d) from Fig. 3.10

Results from the Rotation Analysis Cases (c) and (d)						
Category	Grid cell size (μm)	Expected (mm)	VOF gradient (mm)	PLIC-ASB (mm)	Error VOF gradient (%)	Error PLIC-ASB (%)
Square rotated by 0, Side=151 cells	1	2.416	1.806	2.416	25.241	-0.019
Square rotated by 0.25 (Pi/4), Side=151 cells	1	2.416	2.244	2.48	7.105	-2.628
Square rotated by 0.5 (Pi/4), Side=151 cells	1	2.416	2.097	2.633	13.189	-8.977
Square rotated by 0.75 (Pi/4), Side=151 cells	1	2.416	1.868	2.489	22.663	-3.039
Square rotated by 1 (Pi/4), Side=151 cells	1	2.416	2.313	2.415	4.247	0.04
Semi circle cut out Square rotated by 0, r=70.5, Side=151 cells	1	2.738	2.517	2.771	8.053	-1.202
Semi circle cut out Square rotated by 0.25 (Pi/4), r=70.5, Side=151 cells	1	2.738	2.473	2.82	9.668	-2.981
Semi circle cut out Square rotated by 0.50 (Pi/4) , r=70.5, Side=151 cells	1	2.738	2.36	2.93	13.814	-7.006
				Mean: Std Dev:	13 $\pm$ 8	-3 $\pm$ 3

4.2. Simulating dynamic spherical interfaces with the VOF gradient and PLIC-ASB

For the following test items the general configuration in Section 3.3, Fig. (3.10) type (a) and (b) depict the arrangements used. Based on the objectives of Section 1.1.2, this test is part of Objective 3 (the adiabatic case) followed by Objective 4 (the non-adiabatic case).

4.2.1. Temperature Independent Interfacial Mass Flux

Figure (4.1) is a plot of the anticipated bubble radius as determined theoretically, by the PLIC-ASB method, and by the VOF gradient method. Functionally they all recover the same behaviour, a linear growth rate. The degree of agreement is the only source of conflict. The amount of growth predicted by theory is the initial 100 μm site will grow to about 165 μm. The VOF Gradient has a relative error of 7% compared with the 0.2% of the PLIC-ASB method.

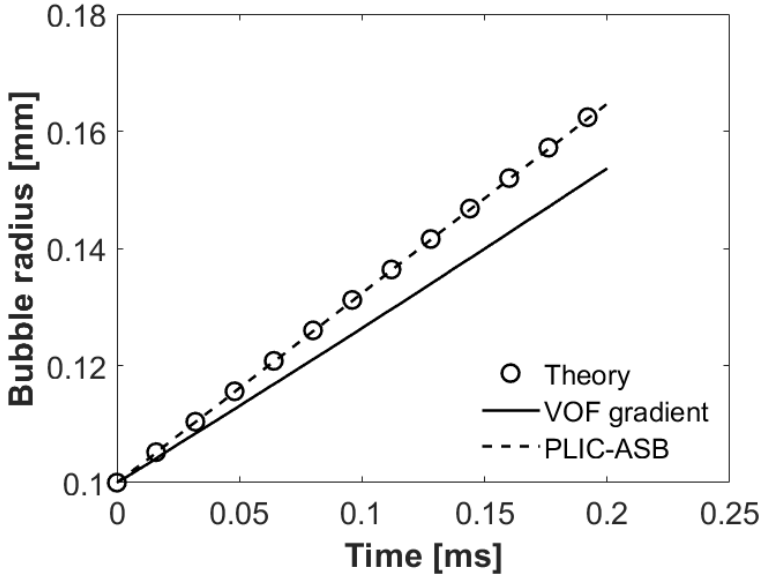

Figure 4.1: Comparison of bubble growth rate with constant mass flux. Surface area estimated with VOF gradient, PLIC-ASB, and theoretical methods. Adapted from prior work [106].

In the previous figure it was shown that the PLIC-ASB had the faster growth rate. In Fig.(4.2) close inspection reveals deformation of the VOF gradient interface. There is a two-fold problem here. The deformation can introduce parasitic velocities and reduce

mass transfer rate by impeding sharp coherent precise calculations of the interface.

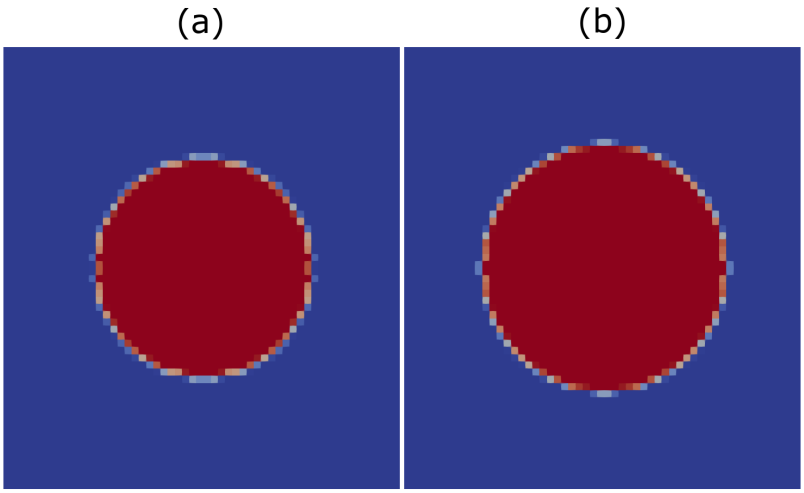

Figure 4.2: Contour of vapor volume-fraction at 0.2 ms. Vapor fraction ranges from one to zero along the color spectrum red to blue. Interface size calculation method: (a) VOF gradient method, (b) PLIC-ASB method. Adapted from prior work [106].

Figure 4.3 shows the velocity vectors obtained with the two analyzed approaches. A velocity jump occurs at the interface due to the mass transfer process. The vapor velocity should be close to zero due to the absence of gravity effects. A vapor expansion occurs (due to the difference in density between the vapor and liquid phases), sending liquid away from the interface. A comparison between the results in Figures 4.3(a) and 4.3(b) shows a reduction in the liquid velocity magnitude with the VOF gradient approach. The smaller velocities are due to the reduced mass transfer rate associated with the under-calculation of the interface surface area while the VOF gradient method is applied. Figure 4.3(a) indicates that the interface-smearing observed increases the vapor velocities, which implies that applying the VOF gradient leads to parasitic velocities. Figure 4.3(b) indicates that the vapor velocity remains close to zero by esti-

mating the interface surface area with the proposed approach. Therefore, results reveal that a proper way of accounting for the interface length with the PLIC-ASB method contributes to achieving multiphase simulations with near-zero parasitic velocities.

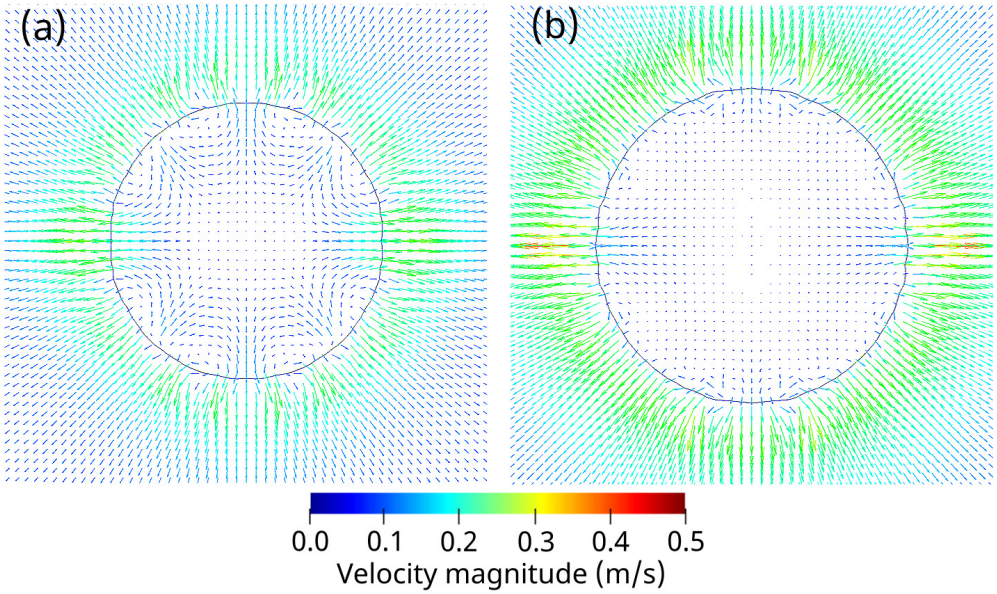

Figure 4.3: Velocity vectors at 0.2 ms. Interface surface area estimated with: (a) VOF gradient method, (b) PLIC-ASB method. The anticipated zero velocities within the bubbles are depicted by the darkest blue shortest vectors, whereas the green vectors in the left and right sides of bubble (a) reveal the parasitic velocities. Adapted from prior work [106].

The total differences here are remarkable: When compared with the theoretical values, in this adiabatic bubble case the relative error was found to be 7% vs. 0.2% for the VOF gradient versus PLIC-ASB respectively. The matter of maintaining a sharp interface has also been demonstrated, though not as soundly to casual observation. These results are found after refining the use of the VOF gradient technique for sharpness, using the gradient technique that suits it best, and eliminating interface degradation as much as possible with in-house UDFs previously mentioned. Considering that this technique is

well established in the literature, and yet mid range values of velocities were shown in the vapor region of the bubble, this level of deficiency was a surprising result.

4.2.2. Temperature Dependant Interfacial Mass Flux

This report continues the spherical bubble arrangement as address previously. The general configuration in section 3.3, Fig. (3.10) type (b) depict the arrangement. Based on the objectives of section 1.1.2, this test is part of Objective 4.

In bubble growth with heat transfer, parasitic velocities near the interface might affect the temperature distribution and the numerical stability. In addition, smaller computational grid-cell sizes are needed to capture the temperature near the bubble edge adequately. For these simulations the VOF gradient and the PLIC-ASB method were used to calculate the interface surface area, as has been the practice in the other comparison test/simulations all other aspects of the computational/experimental setup are kept as identical as possible between tests for each interface method.

These simulating conditions consisted of water at 1 atm as the working fluid with a superheat level of 5 K. The initial bubble radius of 0.1 mm, and the bubble growth time was 0.18 ms with an expected final bubble radius of 0.16 mm. We performed a mesh sensitivity analysis with 1, 0.6, and 0.2 μm grid cell sizes. The courant number was kept below 0.25. The physical simulation time ranged from 3 to 48 hr with a

supercomputer consisting of 48 cores at 3.02 GHz [92]. The highly variable cpu clock time range is due to the nature of a shared resource not a resonable estimate of the code run time. For more information on the high performance computing resource, see section 3.3.3.

Figure 4.4 shows two plots detailing the bubble radius as a function of time, the left is a plot of data from a simulation done with the VOF gradient method. The right is plotted from PLIC-ASB method simulation data. These plots are further stratified by a mesh size and have the theoretical curve places for comparative analysis.

The general curve is well fit by all simulations without any functional deviations in any of the simulations. A more detailed view reveals that the VOF gradient method seems to have a rather good result, but upon inspection it is the instability of this method's non-converging results that just happened on a good result. To the opposite, the PLIC-ASB progresses toward the theoretical ideal as the mesh is refined.

Improved accuracy with refined mesh, within reason, is an expected result for a properly working simulation. This result was not easily shown with static cases without using many orders of magnitude to overcome the noise between mesh size. These data can seem a bit contrived.

The reason why mesh is not refined without limit should be, for most operations, diminishing returns are found. The time to simulate can also become exceptionally long

while maintaining a level of improvement, however factors from the total size of the simulation space to the length of time step are balanced to choose the mesh size and time steps, such as the Courant number. The Courant number is the fraction (or number) of computational cell that the interface transverses per time step.

It is noted here that the reduced size (or higher density) means there is an increasing number of computational cells in the thin thermal film region near the bubble edge. Results show that the VOF gradient method with 1 μm grid cell size underestimates the bubble growth rate, results with 0.6 μm get very close to the theoretical results, and results with 0.2 grid cell size lead to an unexpectedly faster bubble growth rate.

Results in Table 4.1 indicated an underestimation of the interface surface area in the VOF gradient method and an increase in the error as the grid was refined. Therefore, the results in Figure 11 suggest that the VOF gradient method creates a parasitic mechanism that increases the bubble growth rate with the grid refinement (the parasitic velocities adversely advance the interface; see advection[1] term in Eq.(3.14) in section 3.3.2).

The data from the simulations using the PLIC-ASB method show more normative clear trends in keeping with the expectations. These include the trend for standard deviation to account for at least a signifigant portion of the error distribution, the trend for im-

[1]Sub-discipline nomenclature: the advective term is the spatial or nabla/ divergence and the so called unsteady term is the temporal partial.

proving accuracy as the mesh is refined. The PLIC-ASB method shows evidence of converging to the theoretical bubble growth rate. This is the appropriate functionality for a systematically sound numerical method. In a similar way it should be understood that the lack of convergence is a problem as found with the VOF gradient method. Wen et al. covers in the appendix [132] a description of the potential suitability of the VOF gradient method and the need to use a more complex method without the strict mass conservation due to the VOF gradient convergence problem. The conflicting systematic over calculation in static sizes and under calculation on growth rates may lead to it balancing out in some scenarios, but it is a matter of being correct for the wrong reason. As stated early on in this document, such a simulation is a form of animation.

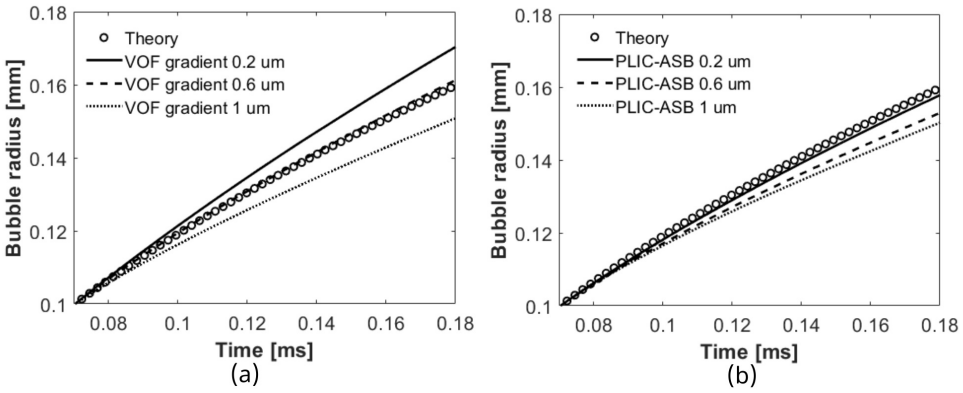

Figure 4.4: Comparison of bubble growth rates at various grid cell sizes obtained with simulations that apply the VOF gradient and PLIC-ASB methods. Adapted from prior work [106].

Figure 4.5 shows the vapor volume fractions at 0.17 ms for three conditions:

(i) VOF gradient method with a grid cell size of 0.6 μm,

(ii) VOF gradient method with a grid cell size of 0.2 μm, and

(iii) PLIC-ASB method with 0.2 μm.

In all these cases, the simulations define a sharp interface. Computational cells with volume fractions between one and zero (fractional value) indicate a continuing or emerging interface cell. When a cell is a composite of vapor and liquid phases it is either due to an interface dividing said cell, a type of mixing/disruption/flotsam jetsam (not found in these simulations), or an error. The simulation with 0.6 μm, (i) shows the larger interface-cells, whereas simulations with 0.2 μm, (ii) and (iii) show extremely small interface-cells where simulating a sharp interface is much more challenging. Consider there are now nine cells in simulations (ii) and (iii) occupying the space of one in simulation (i). Results indicate that the interface remains sharp with both analyzed methods. The interface remains sharp because the simulations used the method proposed by Perez-Raya and Kandlikar designed to preserve the interface sharpness [84, 85]. These results show that the VOF gradient method may predict a significantly more oversized bubble than the PLIC-ASB method for the same grid cell size

Figure 4.6 shows velocity vectors during bubble growth at 0.17 ms for grid cell sizes of 0.6 μm with the VOF gradient method and 0.2 μm with the PLIC-ASB method. Simulations show that during bubble growth, the liquid travels outwards due to the increase in volume when the liquid becomes vapor (the increase in volume comes from

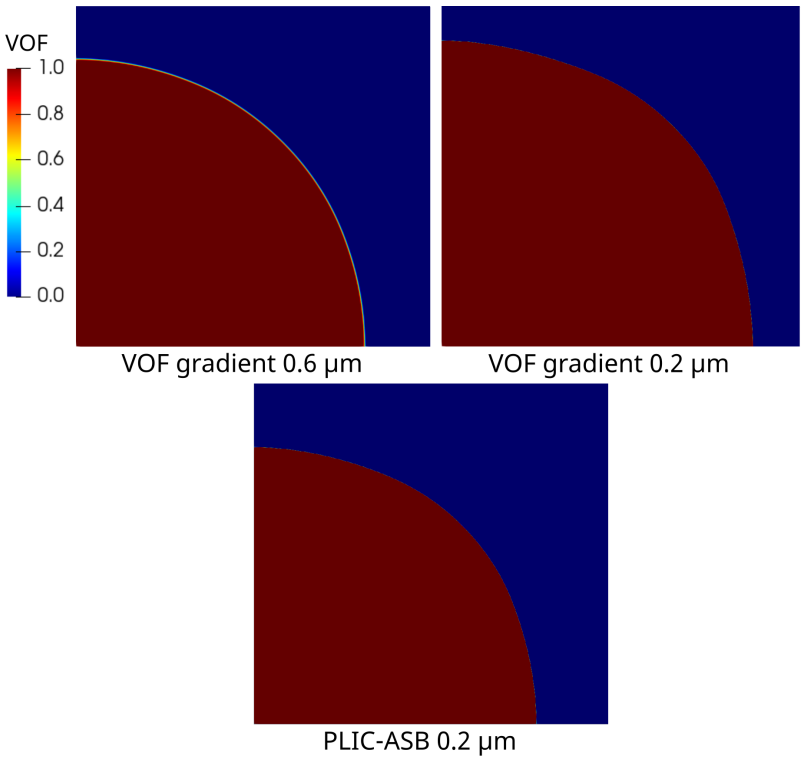

Figure 4.5: This figure depicts a comparison of volume fractions for the computational domains of the bubbles at 0.17 ms obtained with simulations that apply the VOF gradient and PLIC-ASB methods. From symmetry about the x and y axes these are shown as quarter circle representations of the simulated spheres. Adapted from prior work [106].

the fact that the vapor density is much lower than the liquid density). Also, the figure shows fluid going outwards because it ignores the relative motion between the interface velocity and the liquid velocity (the interface velocity is higher than the liquid velocity). The vapor velocity is expected to be equal to zero since the liquid does not expand, and there are no gravity effects considered in the process. The maximum observed velocity magnitude is 0.75 m/s occurring near the axisymmetric horizontal axis. Simulations with the VOF gradient method lead to much higher liquid velocities than expected values. Results show that simulations considering the VOF gradient method

create unrealistic vapor velocities and that such unrealistic velocities become stronger as the computational grid cell size gets refined. A closer look shows that the parasitic vapor velocities have a direction pointing towards the interface. The parasitic velocities might cause the unexpected faster bubble growth rate with finer grids observed in Fig. 4.4(a). The parasitic velocities significantly contribute to the transport of the volume fraction and help the interface move adversely faster. Results in Fig. 4.6 show that the accurate estimation of the interface surface area with the PLIC-ASB method leads to vapor velocities magnitudes near zero, which indicates minimal parasitic velocities. Therefore, results show that the PLIC-ASB approach helps to improve the accuracy of the simulation by suppressing the appearance of parasitic velocities, which allows an interface displacement driven solely by mass transfer interaction mechanisms. The results provide evidence of the relevance of an accurate definition of mass transfer in suppressing the appearance of parasitic velocities and leading to an accurate bubble growth rate.

Table 4.4 shows the calculated mean absolute error (MAE) of the simulated fluid velocity with respect to the theoretical values (see Eq. (3.24)). The MAE was computed for the VOF gradient and the PLIC-ASB methods at various times (in the range of 0.09 to 0.19 ms) throughout the bubble growth process. Results show that the VOF gradient method generates MAE values of the order of 0.1, which are significantly larger relative to the values of 0.03 obtained with the proposed PLIC-ASB method. In

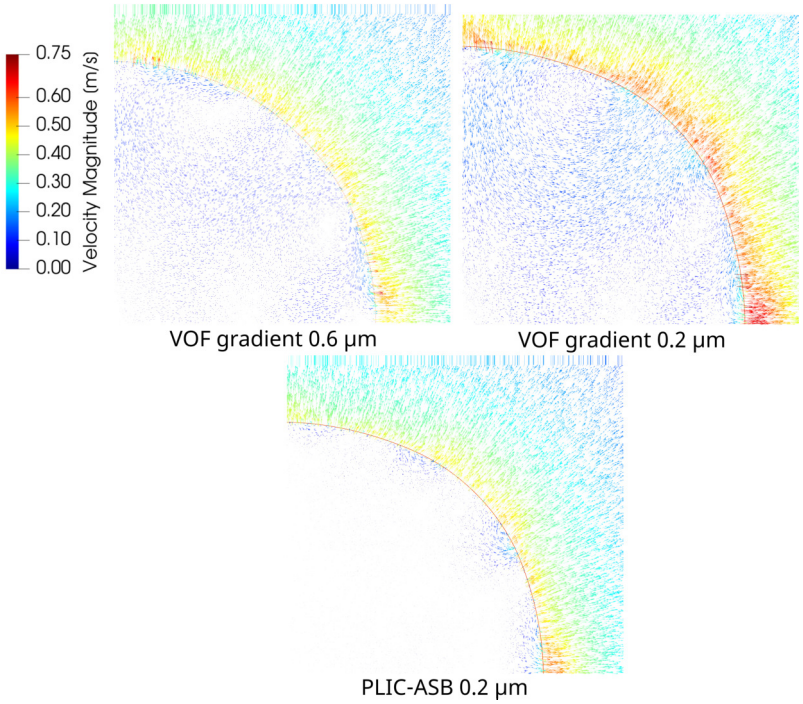

Figure 4.6: This figure depicts the fluid velocity vectors at 0.17 ms obtained with simulations that apply the VOF gradient and PLIC-ASB methods. From symmetry about the x and y axes these are shown as quarter circle representations of the simulated spheres. Adapted from prior work [106].

addition, the simulation with the VOF gradient method generates MAE that increase with time, which indicates that the estimation of interface surface area with the VOF gradient method leads to propagating errors. The simulation with the proposed PLIC-ASB approach shows MAE values that remain almost constant throughout the bubble growth process, which shows that the proposed approach (that precisely calculates the interface surface area) contributes to a more accurate and stable fluid velocity behavior.

Figure 4.7 shows the temperature distribution during the expanding bubble at 0.17 ms. The vapor phase remains at a saturation temperature of 373.15 K, whereas the liquid

Table 4.4: Table comparing the MAE values over the course of simulation time for the VOF gradient and the PLIC-ASB method.

Time (ms)	MAE (VOF gradient) (m/s)	MAE (PLIC-ASB) (m/s)
0.09	0.0727	0.0242
0.12	0.0953	0.0257
0.14	0.1033	0.0259
0.17	0.1155	0.0224
0.19	0.1209	0.0283

phase has a temperature that changes from 373.15 K to 378.15 K. Initially, the domain consists of only liquid at 378.15 K. At zero seconds, a small bubble appears with an interface at the saturation temperature of 373.15 K. The interface cools down the liquid creating a thin thermal film near the interface. Deviations in the velocity of the liquid affect the temperature distribution due to convective transport. In the present case, the high liquid velocities obtained with the VOF gradient method and 0.2 μm grid cells translate into a faster thermal film transport. Also, results show significant deformations of the thermal film with 0.2 μm grid cell size due to the unrealistic velocities generated by the VOF gradient method to estimate the interface surface area. The simulation with 0.6 μm and the VOF gradient method leads to more accurate temperature distributions (similar to the temperature obtained with 0.2 μm and the VOF gradient method). However, with 0.6 μm, significant thermal film deformations are still observed near the corners. The temperature distributions in Fig. 4.7 show that the PLIC-ASB method helps the simulation software to generate simulated thermal films with minimal deformations. Also, the proposed method allows a more realistic mass transfer modeling with a precise estimation of the interface surface area.

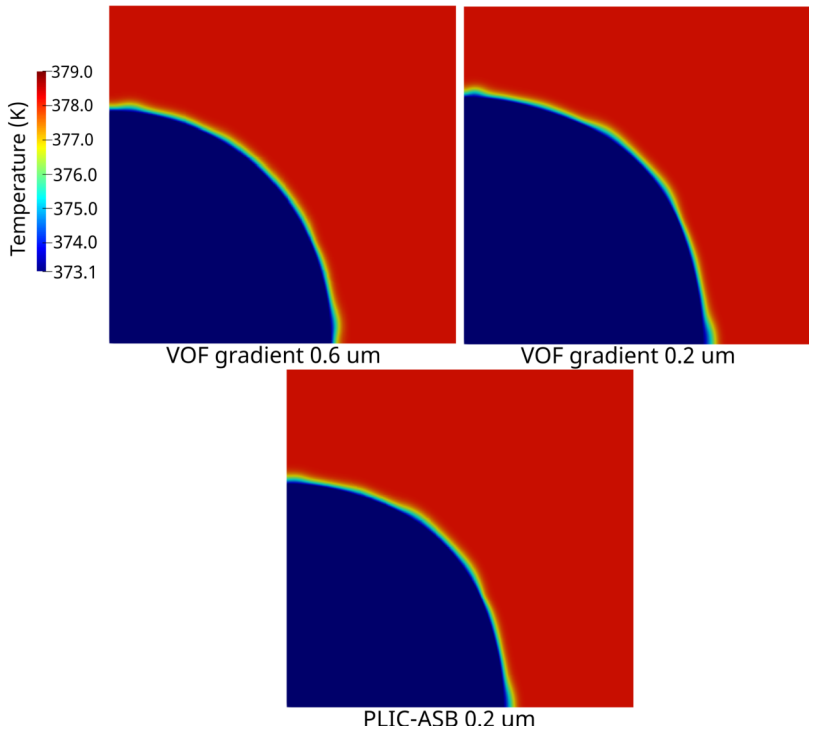

Figure 4.7: These images depict a comparison of temperature distribution at 0.17 ms obtained with the simulations that apply the VOF gradient and PLIC-ASB methods. From symmetry about the x and y axes these are shown as quarter circle representations of the simulated spheres. Adapted from prior work [106].

The converging to an anticipated trend found in the mesh sensitivity test and low relative/low intra-set deviations of noise found in the MAE evaluation are further evidence against systemic error being present in the PLIC-ASB, thereby satisfying the hypothesis both in error reduction and in Design standards, improving size accuracy calculations, exceeding the reduction in the error rates of the existing method, and in finding agreement with theoretical values.

4.3. Simulating Dynamic Bubble Growth Over a Heated Surface with the VOF Gradient and PLIC-ASB

As this report continues the spherical bubble arrangement in this potion is modified to incorperate a heated surface. Unlike the previous version there is now only one axis of symmetry. For a depiction see Fig. 3.11 in section 3.4.1. This test case is part of Objective 4 because along with the non-adiabatic transfer condition, this is the explicit heated wall simulation and empirical results are used to evaluate the simulation results.

The present section shows the results of simulating bubble growth over a heated surface. The simulation considered water as the working fluid at atmospheric conditions. The wall superheat was 6.2 K with a contact angle of 40°. The initial conditions considered a small patch of a spherical bubble with a radius of 0.1 mm. Also, the initial conditions included a linear temperature profile changing from 379.6 K at the heated surface to 373.15 K at 1 mm from the heated surface. Son et al. [111] adopted similar initial conditions. The simulation accounted for microlayer evaporation with the theoretical model reported by Perez-Raya and Kandlikar [85]. The microlayer evaporation rate was defined at the computational cell that contains the triple-phase contact line (solid-liquid-vapor). Ansys-Fluent was customized to preserve the interface sharpness, define mass transfer at the interface, and account for the interface saturation temperature. Perez-Raya and Kandlikar [84] give a detailed description of the methods applied to customize Ansys-Fluent. The

simulations in the present work considered the VOF gradient and the proposed PLIC-ASB method to compute the interface-surface area in the computational cells. The simulations considered a grid cell size of 12 um with a variable time step size with a constant Courant number equal to 0.2.

Figure 4.8 shows the change of the temperature distribution and bubble shape as a function of time for simulations considering the VOF gradient and the proposed PLIC-ASB method. The left sub-panes show the results obtained with the VOF gradient method and the right sub-panes show the results obtained with the PLIC-ASB method. Results show that both approaches can deal with the interface saturation condition. Although the VOF gradient leads to less accurate bubble growth rates, the error does not propagate to the temperature distribution near the contact line. These results imply that possible parasitic velocities appearing with the VOF gradient approach may be absorbed by the strong evaporation occurring at the microlayer. Similarly, the possible parasitic velocities are not significant enough in this configuration to seriously distort or break the interface. However there is a observed difference in the bubble shape which indicates that the evaporation outside the microlayer contributes to the bubble growth rate to some degree and that simulations without the VOF gradient problems (e.g. the proposed PLIC-ASB method) lead to a faster bubble growth rates.

The simulation was based on an existing experiment in the liturature by Siegel and Keshock [108] so as to compare with empirical results. The data is given here in Fig. 4.9 to show

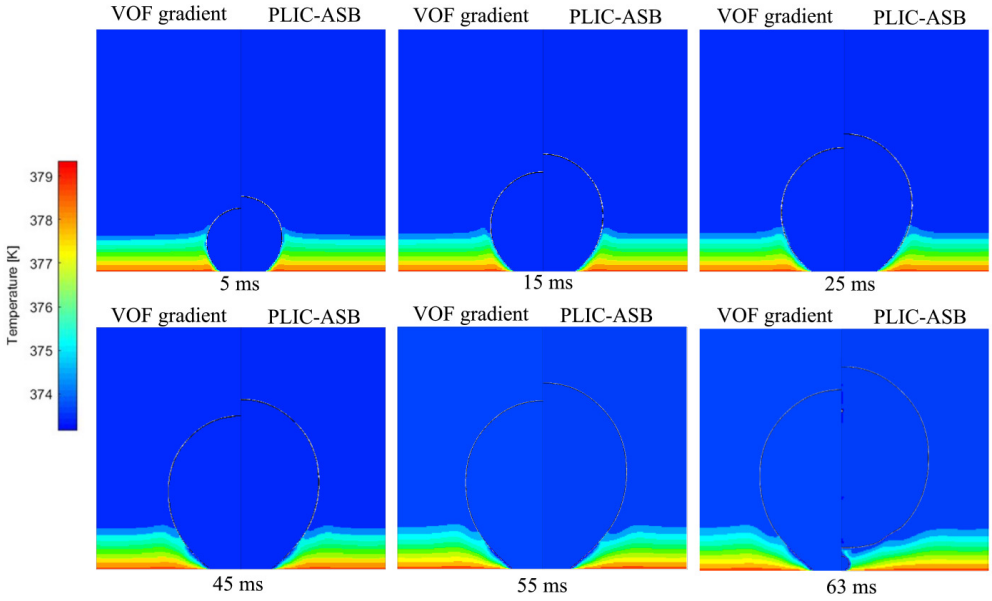

Figure 4.8: A temperature distribution map. The temperature variation during a bubble cycle (first bubble) obtained with simulations that apply the VOF gradient (left sub-panes) and PLIC-ASB (right sub-panes) methods. The final bubble departure for the PLIC-ASB simulation is shown in the lower right panel, having occurred slightly before 63ms.

that repeated experiments showed some variability and to explain the subsequent steps.

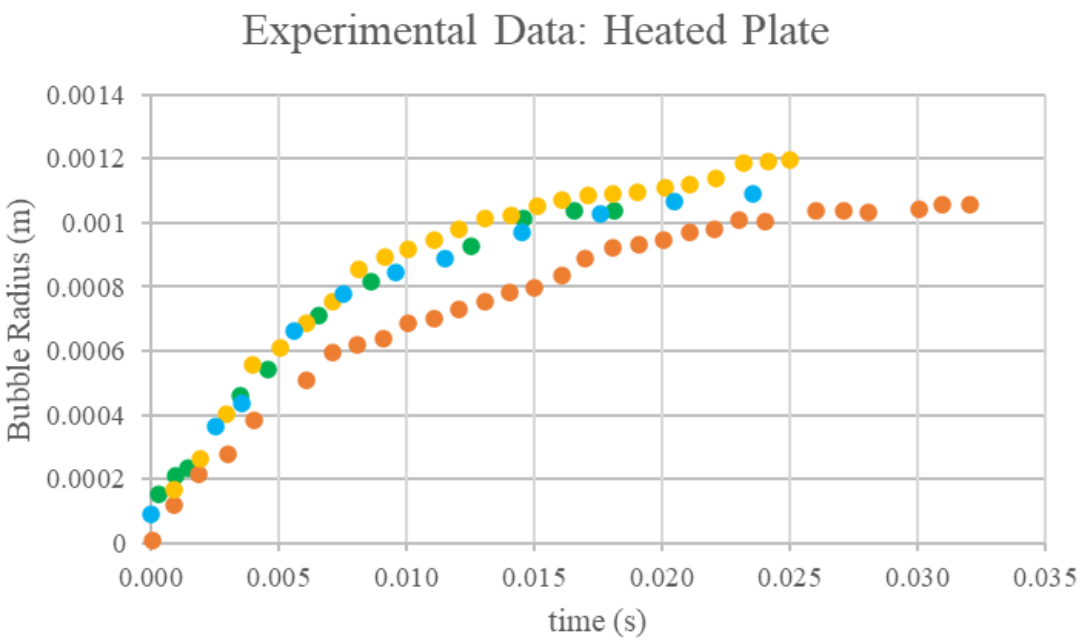

Figure 4.9: Experimental data reported by Siegel and Keshock [108].

The multiple data was converted into a combined data set. Of note was experimental error. The measurement on the bubble nucleation (start time), radius, and time precision in particular. Matters of the temperature nucleation site control degassing are neglected due to lack of knowledge on those details. The initial nucleation and time precision can be additive as an x-error for time. The y-error of radius comes from optical cross bubble measurement of the diameter whereas the data in this report is the result of assuming the bubble is spherical and calculating the radius. Due to this the simulation-experiment data is diverging such that the simulation is reporting low radius and the experiment is reporting high radius.

For the time error as it is possible for at least one frame in a camera reel to be off that would be the minimal, but it is also not too impossible that the bubble starts and is missed

or that there is a frame or two off the perfect timing as the timing control as listed was not integer steps of frames about 3500 per second. which sounds unimpeachable, but two frames off is 0.00057s which is over a tenth of a major gridline division on the plots listed here.

The amalgamated data needed a clear rubric of fit or fail. To this end the best theoretical approach was used as a formulation for a fit that remained data driven in that similar to a moving average a distribution confidence interval was applied (see Fig. 4.10). If the fit was only one of either of these it would be less helpful as the non uniform data sampling from the experiment provides chaotic results from pure histogram style distribution fitting. If the fit was only a theory based fit that most closely coincided with the data it would not have a range for uncertainty. The specific thory is from the expirimental publication where different recommendations on fractional values of t, the x-axis, were recommended. This $c1t^0.5 - c2t$ (simplified form) provided the simplest fit while not curving down at the end, requiring strange initial conditions, or clearly deviating from the data. The error bars are depicting the 95% confidence interval for the data to agree with the fit. The Fit itself is a dashed line most listed in the legend of the following plot, but slightly obfuscated to reinforce that the center estimate of the fit is not a perfect fit any more than some other location on the error bar. This slightly nebulous approach is because the data is a bit limited and it is possible for a new data series done under the same conditions with unexpected results to arrive that alters the understanding of the fit.

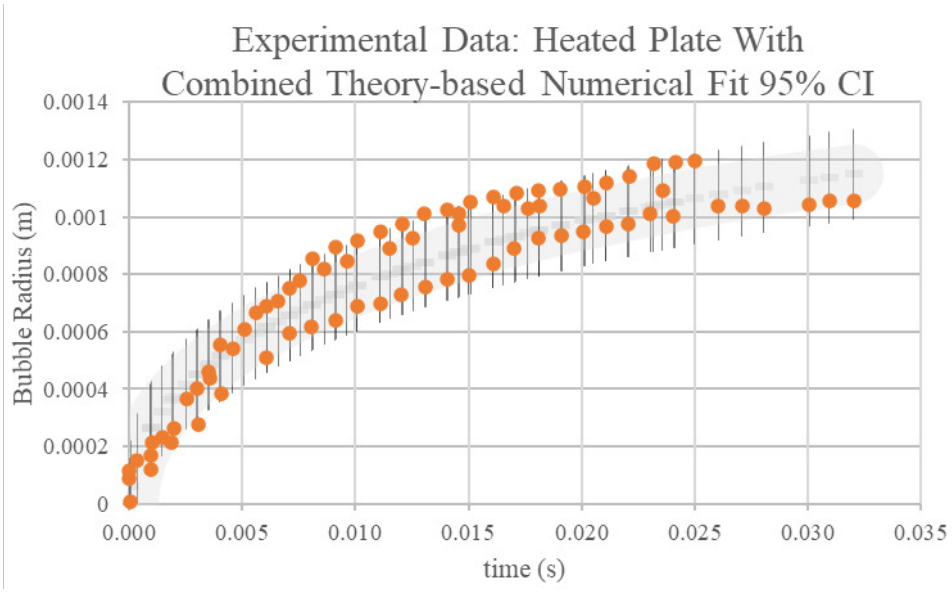

Figure 4.10: Theory based fit applied to Siegel and Keshock's data [108].

Figure 4.11 is a plot of the experimental data of the bubble growing over a heated plate with the previously described fit and the simulated data. The plot shows the bubble radius as a function of time. The PLIC-ASB method shows bubble growth using this method outpaces the bubble growth of the VOF gradient method simulations. Considering the entire curve the 4μm PLIC-ASB method remains within the 1-sigma error limit. This places it in agreement with the experimental data. The 12μm PLIC-ASB method remains within the 1-sigma error limit for much of the plot, developing the most error representing a 5% relative error with the fit boundary. The VOF gradient method 4μm was similar to the 12μm PLIC-ASB in terms of following the sigma error- but its maximal error is 7%. Finally the maximum error in the 12μm VOF gradient was 12% and it mostly was outside of the 1-sigma error range.

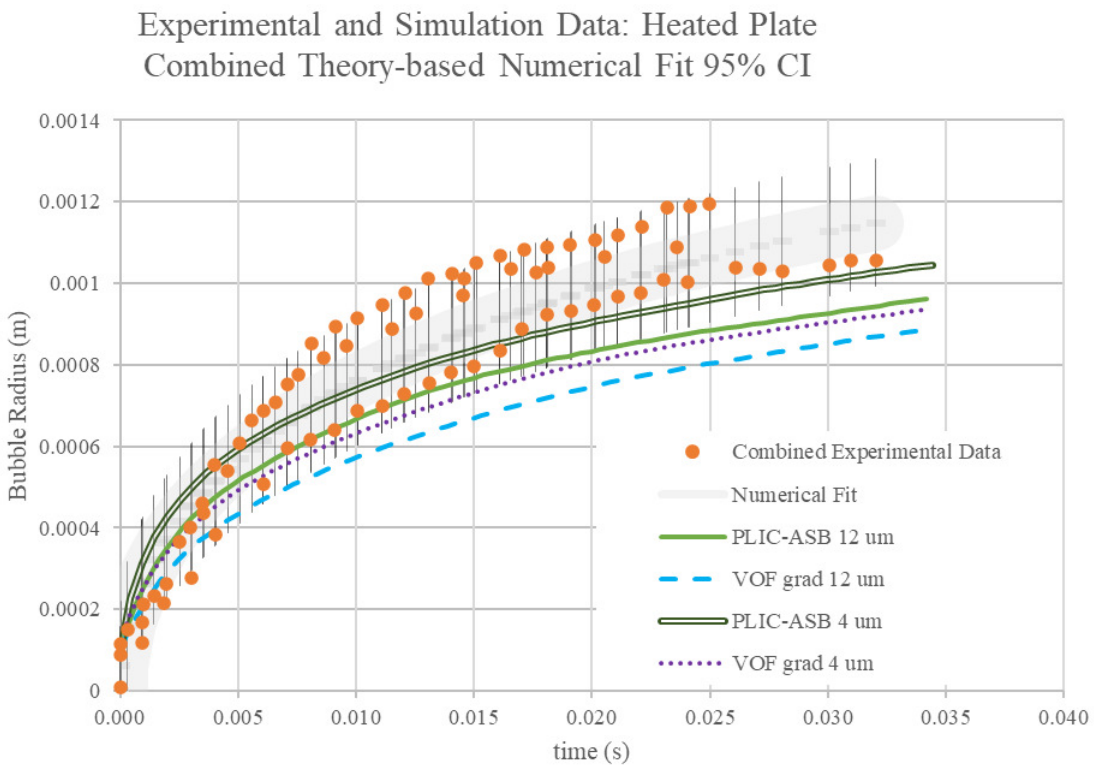

Figure 4.11: Comparing simulations with VOF gradient and the PLIC-ASB methods against available experimental data with theory based based fit. Experimental data reported by Siegel and Keshock [108].

Hypothosis item 4 is: Evidence for consistent improved accuracy will be found. The developed method will agree with theoretical and empirical data, and reduce the VOF gradient error rate significantly (by one half).

Here the PLIC-ASB has made full agreement with the empirical data. Considering the maximum relative error as the maximum deviation from the fit: The PLIC-ASB values are:

12μm = 5%, 4μm = 0%

The VOF gradient values are:

12μm = 12%, 4μm = 7%

This again offers evidence in agreement with the hypothesis item.

4.4. Simulation Time and Error Relationship

In this section the time of simulations is considered. The time required to simulate an experiment is not just an aspect of its computational expense, it can also be a determining factor in the feasibility of performing enough simulations for them to be of use. For example, a simulation that requires a few days to run may cost extra resources, but in theory an intensive parametric study can be performed. The same is not the case for a simulation that takes years. There may be longitudinal studies for certain cases but every month longer a simulation takes the less feasible the simulation is for most researchers.

Previous evaluations noted changes in error levels with reduced cell size. The pattern on display should be the modest accuracy gains and evidence of any systemic error in the evaluated group of methods and techniques. The total time of the simulation is a similar classification and in fact simulation time and the cell size (of a simulation with constant total size) have a fairly direct correlation. For a number of reasons twice as

many cells does not mean twice the calculation time. The time can be slower due to added complexity, exceeding resource draw, and depending on the set up, refining the mesh will either slow the time steps (constant/limited Courant number) or breaking the interface growth rate otherwise.

Due to the variety of processes used and the poor signal to noise ratio of the cluster computing data, there is not an extensive data set. Testing of this sort and more was done but in isolation, that is singling out components and isolating variables. That initial development testing for verification and to ensure the PLIC-ASB was not introduction error into simulations adding helped uncover the reasons for the VOF gradient problems.

Considering the previously detailed spherical bubble growth in superheated liquid , the following plots evaluate the reported simulation-time data from a private computer (Dr. Perez-Raya) with simulation details included below. The expected trend along with the stated mesh correlation is that longer times will provide more calculation and less error. The refined mesh, if properly handled likewise reduces error but increases computation time. In Fig. 4.12 this trend is observed with the PLIC-ASB method, but it is not followed by the VOF gradient simulation method. It is known that this method does not converge properly and this is just evidence of this.

Considering the anticipated trend, it is known that there are error contributions of known and stochastic types. For simplicity we can consider an arbitrary decay function as driving the error directly related the nature of the problem (e.g. curvature, gradient methods) and

these others above as separate factors. If as time extends to an arbitrarily large value there is no resolution, the factors would be constant. In the case of the segregation method there is a geometric relation etc. For these reasons a fit is added in Fig. 4.13 to illustrate a decay function. Here it fits very cleanly, but it is likely that the fit has some error and per this discussion it is hard to justify that the full curvature of a decay function is due to intrinsic error properties. Something like $Rel.Error = C1 + C2(shapegeometry)\, t^{1/3} + C3e^{-C4t}$ where C1, C2, C3, C4 are constants to define the terms above.

Expressing the error as a decay function, with even an unknown base is just intended to help motivate thought on this due to prior experience. hard numbers are left out to be explicit that the data does not support formulating such a function as it would be gross over-fitting to have as many constants as data points.

Simulation details for the simulation of the spherical bubble growth in superheated liquid:

Courant number of 0.15

Converge criteria 1e-06 for all the governing equations (energy and Navier-Stokes)

Computer characteristics speed 3.7 GHz, 10 cores, 64 GB Memory ram

Simulation consider 5 cores in parallel processing

Simulation times with PLIC-ASB and VOF were similar

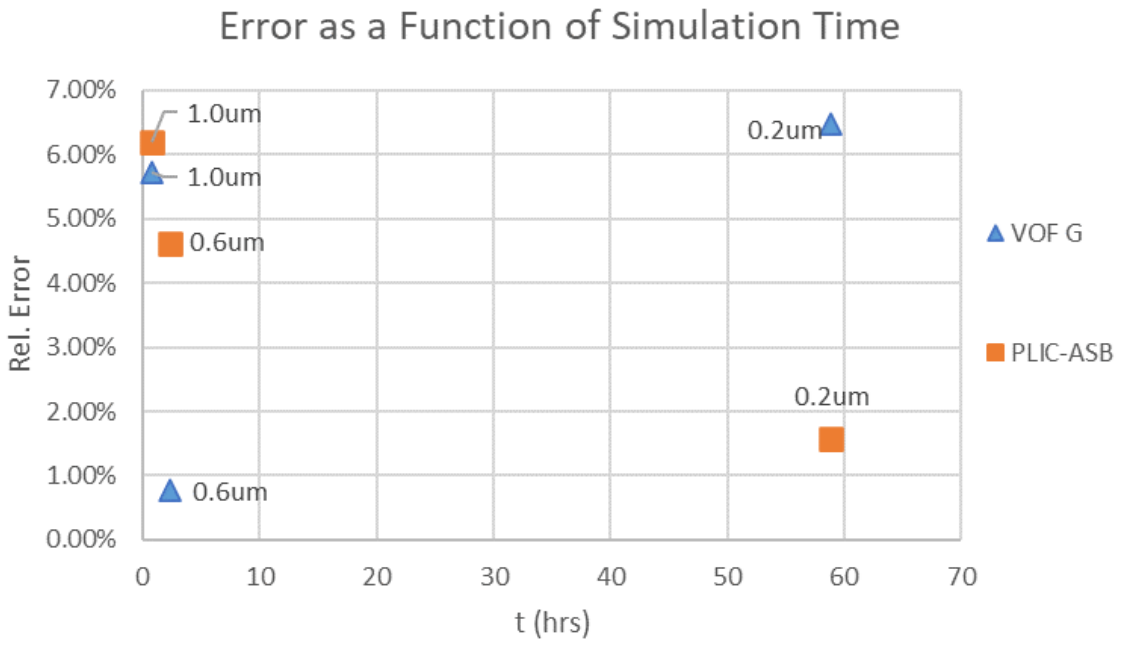

Figure 4.12: Plot of maximum relative error as a function of simulation time. The related simulations are from the spherical bubble growth in superheated liquid. The Orange squares represent the PLIC-ASB method and the blue triangles represent the VOF gradient method.

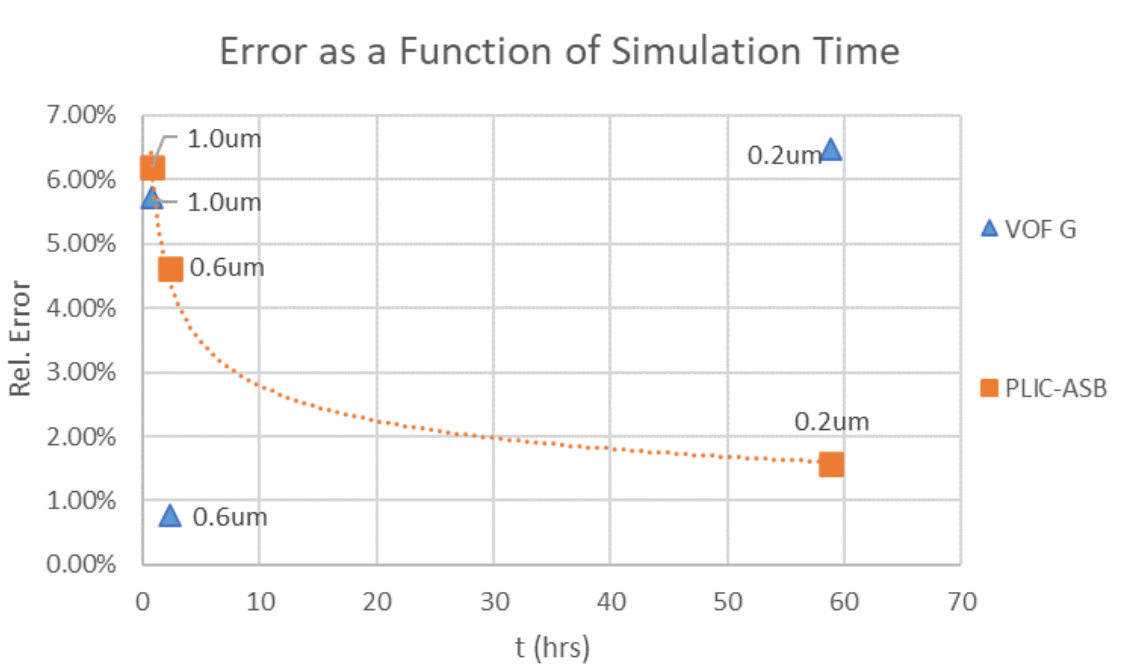

Figure 4.13: Slightly adapted version of the prior plot. This plot has an arbitrary decay function to illustrate a noise-less fit. As there certainly is noise it is reasonable to motivate from this more stability in the mesh. As before, Plot of maximum relative error as a function of simulation time. The related simulations are from the spherical bubble growth in superheated liquid. The Orange squares represent the PLIC-ASB method and the blue triangles represent the VOF gradient method.

The bubble growth over a heated surface simulation is a comparison with true empirical data. In the analysis simulation data was compared with this experimental data by fitting it to a theory informed curve that was otherwise data driven. A pure average would have shown high fluctuation as the experiments did not all have equivalent sampling. The simulation data for the 4μ VOF gradient result was unfinished at the time of this report, so an extrapolation was done. In Fig. 4.14 the maximum error is determined by using the maximal deviation from the confidence interval. This means that the zero error may be improved upon in some sense given future advancements; however, it would be incorrect to give the center value of the fit a special condition as if it were a bullseye. Only rarely did

even the empirical values coincide with the fit center rather than falling above or below.

The trend for the data in this plot is as expected. This simulation can take a prohibitively long time and so these limiting cases with course mesh actually provide data on these challenges. The relative shallow angle of the slope between the data points and the agreement on this with both interface methods is a good sign in that both simulations, the 12μm mesh and 4μm mesh are reasonably stable and not a just statistical noise/ an error fluctuation that happened to push in the right direction. As referenced in the section 4.3 the error associated with the experiment and the known error in the simulation lead to a more favorable view of the results. Apart from the 12μ VOF gradient case, the simulations were either within 1-sigma or just outside it, and even the 12μ VOF gradient case did not near the 3-sigma range.

Simulation details for the simulation of bubble growth over a heated surface in superheated liquid:

Courant number of 0.15

Converge criteria 1e-06 for all the governing equations (energy and Navier-Stokes)

Computer characteristics speed 3.7 GHz, 10 cores, 64 GB Memory ram

Simulation consider 5 cores in parallel processing

Strong evaporation at the contact line leads to fast interface displacements. Keeping the Courant number constant translates into lower time-steps and longer computational times

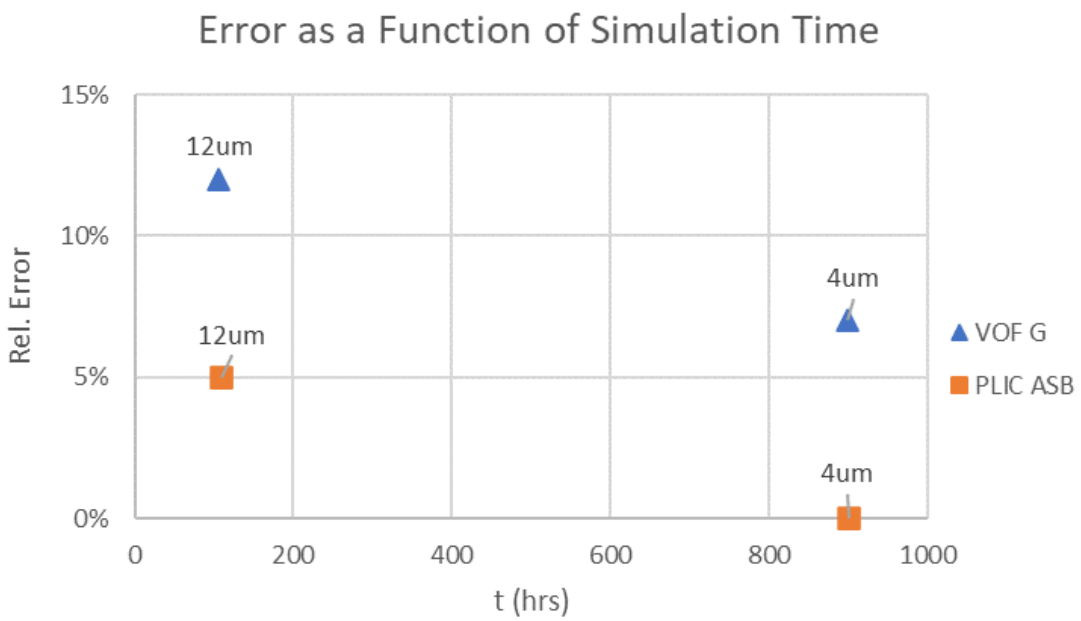

Figure 4.14: Plot of maximum relative error from data driven fit as a function of simulation time. The related simulations are from the bubble growth on a heated surface. The Orange squares represent the PLIC-ASB method and the blue triangles represent the VOF gradient method.

Chapter 5

Conclusions

The present work proposes and describes the piecewise-linear interface calculation analytical size-based (PLIC-ASB) method. The proposed PLIC-ASB method calculates the interface size in multiphase phenomena precisely. The inputs to PLIC-ASB are the normal vector and the cell volume fraction, and the output is the interface size at the computational cell.

Multiphase simulations with mass transfer utilizing the VOF interface tracking method require the computation of the interface length to ensure accurate interface displacements. Researchers have adopted different approaches to compute the mass transfer rate (i) empirical coefficients, (ii) calculation of the interface length with VOF gradients, (iii) interface reconstruction techniques, (iv) level-set method with interface velocities, and (v) other less common alternative approaches. None of the available techniques gives an analytical method to calculate the inter-

face size in the computational cell, which significantly influences the calculation of the mass transfer and the simulation results. The proposed PLIC-ASB gives an analytical algorithm to calculate the interface size, which consists of basic steps (transformation and rotation of the interface). The PLIC-ASB allows accurate and direct computation of mass transfer with temperature gradients at the interface rather than depending on empirical coefficients, assumptions with VOF gradients, or elaborated interface reconstructions. Application of the PLIC-ASB improves the capability of computer modeling of multiphase in capturing the fundamental mechanisms of heat and mass transfer by allowing an accurate computation of mass transfer.

Simulations with the VOF interface tracking algorithm utilizing the VOF gradient method are unable to show convergence of the bubble growth rates with a decrease in the grid cell size (mesh sensitivity analysis). Also, simulations with the VOF method and the empirical coefficients approach are fine-tuned to fit experimental or theoretical growth rates, which prevents it from achieving mesh independence (a coefficient can be determined for each mesh). Only VOF simulations performing interface reconstructions generate convergence of the results to the theoretical bubble growth rates, with an improved accuracy with smaller grid cell size. Unfortunately, these in-house numerical codes do not provide an analytical method to calculate the interface surface area required to compute mass transfer. The present work customized Ansys-Fluent to test and compare the performance of the proposed PLIC-ASB against the commonly adopted VOF gradient method. The successful implementation of PLIC-ASB demonstrates that the proposed approach is compatible with conventional discretization and numerical schemes.

Conclusions

Results indicated that the VOF gradient method is unable to reproduce theoretical spherical bubble growth rates. It was observed that the error in estimating the interface length with the VOF gradient gets larger as the grid cell size decreases. Conversely, simulations with the proposed PLIC-ASB were able to generate the desired bubble growth rates with improved accuracy with smaller grid cell sizes. Simulations with a grid cell size of 0.2 μm generated an error relative to the theoretical values of less than 2%. The PLIC-ASB method allowed a precise mass transfer computation, translating into accurate bubble growth rates and heat transfer interfacial mechanisms. The PLIC-ASB method allowed a precise simulation of spherical bubble growth rates when utilizing customized numerical software, representing a significant breakthrough in multiphase flow modeling.

Bubble growth over a heated surface represents a relevant problem in heat transfer applications. The reason is that boiling gives heat transfer coefficients far greater than single-phase heat transfer processes. It was found that simulations with level-set interface tracking are able to reproduce experimental bubble growth rates with a distinguished accuracy. However, simulations with the VOF interface tracking method face difficulties in achieving precise mass transfer modeling, which prevents proper representation of the heat and mass transfer characteristics. The technical literature lacked a numerical framework applying the VOF method showing good agreement against available experimental bubble growth rates. The software Ansys was customized to simulate nucleate boiling with a sharp interface and with a direct computation of mass transfer based on the VOF gradient method or the proposed PLIC-ASB

approach in the macro-region. Also, a coupled theoretical model to the numerical simulation accounted for microlayer evaporation occurring up to 6 μm from the heated surface. The direct computation allowed a simulation that performs a computation of mass transfer utilizing parameters that come directly from the simulation results (e.g., temperature distribution near the interface) rather than utilizing fine-tuned empirical coefficients. Results showed that the proposed PLIC-ASB method significantly contributes to improving the accuracy of the simulation despite the strong influence of the microlayer evaporation model on the bubble growth process. The results indicate that mass transfer outside the microlayer plays a relevant role in liquid evaporative processes. Also, these results imply that proper modeling of the heat and mass transfer in the macro-region is required to numerically capture the mechanisms of fluid and heat transport. Simulations with an accurate estimation of the mass transfer with the proposed PLIC-ASB get significantly closer to the available experimental results, which allows acceptable reproducibility of the experiments.

The theoretical microlayer model coupled to the numerical simulation appears due to limitations in computer resources required to model conditions below a regular grid cell size of 6 to 12 μm. With advances in computer chip technologies and in methods to generate more effective computer modeling (e.g., adaptive mesh refinement and modeling with physics-informed artificial intelligence), it is anticipated that in the near future numerical modeling of boiling flows will eliminate the requirement of an external theoretical model to capture the evaporative and heat transfer mechanisms at the contact line. It is reasonable to believe that the proposed PLIC-ASB

method will play a vital role in allowing such a transition to highly precise numerical simulations that are independent of theoretical assumptions. PLIC-ASB gives the link required to perform accurate computations of mass transfer at interfaces. Results showed that PLIC-ASB allows simulations capable of generating theoretical solutions with improved accuracy as the grid gets finer. Also, it was found that PLIC-ASB contributes to improving the accuracy of nucleate boiling simulations. One of the main advantages of the proposed approach comes from its composition, consisting of only a few steps to give the outcome of the interface size in the computational cells. Such a non-sophisticated structure allows its implementation in available numerical software, which opens the door for future researchers to develop numerical analysis without requiring advanced in-house codes that only experts in interface tracking and numerical methods can access. As a result, PLIC-ASB contributes to increasing the research to be done in the field of multiphase simulations with heat and mass transfer, which will allow to development of devices required for advanced multiphase technological applications.

Bibliography

[1] Population Properties of Compact Objects from the Second LIGO–Virgo Gravitational-Wave Transient Catalog.

[2] International Roadmap for Devices and Systems (IRDS™) 2022 Edition - IEEE IRDS™, 2022.

[3] Resources Archive - Semiconductor Industry Association, 2022.

[4] Semiconductor Industry Association, February 2023.

[5] Marieke Ahlers, Alexander Buck-Emden, and Hans-Jörg Bart. Is dropwise condensation feasible? A review on surface modifications for continuous dropwise condensation and a profitability analysis. *Journal of Advanced Research*, 16:1–13, March 2019.

[6] Mohammad W. Akhtar and Stanley J. Kleis. Boiling flow simulations on adaptive octree grids. *International Journal of Multiphase Flow*, 53:88–99, July 2013.

[7] Eduardo Aktinol and Vijay K. Dhir. Numerical Simulation of the Effect of Contact Angle on the Thermal Response of the Solid During Nucleate Pool Boiling. *Interfacial Phenomena and Heat Transfer*, 2(4), 2014.

[8] Eduardo Aktinol, Vijay K. Dhir, Talbot Jaeger, and Walt Mirczak. Numerical simulations of nucleate boiling and film condensation heat transfer in a fixed-volume chamber in microgravity. *International Journal of Heat and Mass Transfer*, 132:886–905, April 2019.

[9] SG Bankoff, WJ Colahan Jr, and DR Bartz. Summary of Conference on Bubble Dynamics and Boiling Heat Transfer Held at the Jet Propulsion Laboratory, June 14 and 15, 1956. Technical report, California Inst. of Tech., Pasadena. Jet Propulsion Lab., 1956.

[10] Raunak Bardia and Mario F. Trujillo. Assessing the physical validity of highly-resolved simulation benchmark tests for flows undergoing phase change. *International Journal of Multiphase Flow*, 112:52–62, March 2019.

[11] Vatsal Jayantilal Bhuva, Jash Pranav Jani, Abhay Patel, and Nishant Tiwari. Effect of bubble coalescence on two-phase flow boiling heat transfer in raccoon microchannel - A numerical study. *International Journal of Heat and Mass Transfer*, 182:121943, January 2022.

[12] Justin Broughton and Yogendra K. Joshi. Flow boiling in geometrically modified microchannels. *Physics of Fluids*, 33(10):103308, October 2021.

[13] Murat Bulut, Maharshi Shukla, Satish G. Kandlikar, and Nedim Sozbir. Experimental study of heat transfer in a microchannel with pin fins and sintered coatings. *Experimental Heat Transfer*, 0(0):1–16, February 2023. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/08916152.2023.2176566.

[14] Lubomír Bureš and Yohei Sato. Direct numerical simulation of evaporation and condensation with the geometric VOF method and a sharp-interface phase-change model. *International Journal of Heat and Mass Transfer*, 173:121233, July 2021.

[15] Yuanwei Cao and Rafael Macián-Juan. Numerical investigation of vapor bubble condensation in subcooled quiescent water. *Nuclear Engineering and Design*, 388:111621, March 2022.

[16] Zhizhu Cao, Dongliang Sun, Bo Yu, and Jinjia Wei. A coupled volume of fluid and level set method based on analytic PLIC for unstructured quadrilateral grids. *Numerical Heat Transfer, Part B: Fundamentals*, 73(4):189–205, April 2018.

[17] Fucheng Chang, Yuhao Shang, He Hu, Xi Li, Kaikai Guo, and Huixiong Li. Numerical study on magnetic nanofluid (MNF) film boiling in non-uniform magnetic fields generated by current carrying wires. *International Journal of Thermal Sciences*, 175:107461, May 2022.

[18] Heung June Chung and Hee Cheon No. Simultaneous visualization of dry spots and bubbles for pool boiling of R-113 on a horizontal heater. *International Journal of Heat*

and Mass Transfer, 46(12):2239–2251, 2003.

[19] LIGO Scientific Collaboration and Virgo Collaboration. Observation of Gravitational Waves from a Binary Black Hole Merger. *Physical Review Letters*, 116(6):061102, February 2016.

[20] M. G. Cooper and A. J. P. Lloyd. The microlayer in nucleate pool boiling. *International Journal of Heat and Mass Transfer*, 12(8):895–913, August.

[21] Bart J. Daly. Numerical Study of Two Fluid Rayleigh-Taylor Instability. *The Physics of Fluids*, 10(2):297–307, February 1967. Publisher: American Institute of Physics.

[22] Bart J Daly. A technique for including surface tension effects in hydrodynamic calculations. *Journal of Computational Physics*, 4(1):97–117, June 1969.

[23] Vijay K. Dhir, Gopinath R. Warrier, and Eduardo Aktinol. Numerical Simulation of Pool Boiling: A Review. *Journal of Heat Transfer*, 135(6):61502, 2013.

[24] VK Dhir. Nucleate and transition boiling heat transfer under pool and external flow conditions. *International journal of heat and fluid flow*, 12(4):290–314, 1991.

[25] M. Dianat, M. Skarysz, and A. Garmory. A Coupled Level Set and Volume of Fluid method for automotive exterior water management applications. *International Journal of Multiphase Flow*, 91:19–38, May 2017.

[26] William S. Duff and David A. Hodgson. A simple high efficiency solar water purification system. *Solar Energy*, 79(1):25–32, July 2005.

[27] Ronald P. Fedkiw, Tariq Aslam, Barry Merriman, and Stanley Osher. A Non-oscillatory Eulerian Approach to Interfaces in Multimaterial Flows (the Ghost Fluid Method). *Journal of Computational Physics*, 152(2):457–492, July 1999.

[28] Stavros Fostiropoulos, George Strotos, Nikolaos Nikolopoulos, and Manolis Gavaises. Numerical investigation of heavy fuel oil droplet breakup enhancement with water emulsions. *Fuel*, 278:118381, October 2020.

[29] Benjamin Franz, Axel Sielaff, and Peter Stephan. Numerical Investigation of Successively Nucleating Bubbles During Subcooled Flow Boiling of FC-72 in Microgravity. *Microgravity Science and Technology*, 33(2):27, March 2021.

[30] Deepak Garg and V. K. Dhir. A Unified Three-Dimensional Numerical Model for Boiling Curve in a Temperature Controlled Mode1. *Journal of Heat Transfer*, 141(1), November 2018.

[31] Faroogh Garoosi and Kamel Hooman. Numerical simulation of multiphase flows using an enhanced Volume-of-Fluid (VOF) method. *International Journal of Mechanical Sciences*, 215:106956, February 2022.

[32] A. Gholijani, C. Schlawitschek, T. Gambaryan-Roisman, and P. Stephan. Heat transfer during drop impingement onto a hot wall: The influence of wall superheat, impact velocity, and drop diameter. *International Journal of Heat and Mass Transfer*, 153:119661, June 2020.

[33] Frédéric Gibou, Liguo Chen, Duc Nguyen, and Sanjoy Banerjee. A level set based sharp interface method for the multiphase incompressible Navier–Stokes equations with phase change. *Journal of Computational Physics*, 222(2):536–555, March 2007.

[34] Frederic Gibou, Ronald P. Fedkiw, Li-Tien Cheng, and Myungjoo Kang. A Second-Order-Accurate Symmetric Discretization of the Poisson Equation on Irregular Domains. *Journal of Computational Physics*, 176(1):205–227, February 2002.

[35] Ganesh Guggilla, Ramesh Narayanaswamy, Peter Stephan, and Arvind Pattamatta. Influence of flow rate and surface thickness on heat transfer characteristics of two consecutively impinging droplets on a heated surface. *International Journal of Heat and Mass Transfer*, 165:120688, February 2021.

[36] D. Z. Guo, D. L. Sun, Z. Y. Li, and W. Q. Tao. Phase Change Heat Transfer Simulation for Boiling Bubbles Arising from a Vapor Film by the VOSET Method. *Numerical Heat Transfer, Part A: Applications*, 59(11):857–881, June 2011.

[37] Cong-Tu Ha, Sun Youb Lee, and Jae Hwa Lee. Application of a high-order MP scheme to computation of multi-phase flows with heat and mass transfer. *International Journal of Heat and Mass Transfer*, 183:122055, February 2022.

[38] S. Hardt and F. Wondra. Evaporation model for interfacial flows based on a continuum-field representation of the source terms. *Journal of Computational Physics*, 227(11):5871–5895, May 2008.

[39] Eugene Hecht. *Optics*, volume 4. Addison Wesley, San Francisco, CA USA, 2002.

[40] Stefan Herbert, Sebastian Fischer, Tatiana Gambaryan-Roisman, and Peter Stephan. Local heat transfer and phase change phenomena during single drop impingement on a hot surface. *International Journal of Heat and Mass Transfer*, 61:605–614, June 2013.

[41] C. W Hirt and B. D Nichols. Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of Computational Physics*, 39(1):201–225, January 1981.

[42] Mark Ho, Guan Heng Yeoh, John Arthur Reizes, and Victoria Timchenko. Bubble flow simulations using the intersection marker (ISM) interface tracking method. *International Journal of Numerical Methods for Heat & Fluid Flow*, 28(1):118–137, January 2018.

[43] Haotian Huang, Jianhua Fan, Jianquan Lin, Qian Zhao, Yuyue Zhang, and Yimin Xiao. Numerical phase change model considering crystal growth under supercooling. *Applied Thermal Engineering*, 200:117685, January 2022.

[44] Grégory Huber, Sébastien Tanguy, Michael Sagan, and Catherine Colin. Direct numerical simulation of nucleate pool boiling at large microscopic contact angle and moderate Jakob number. *International Journal of Heat and Mass Transfer*, 113:662–682, October 2017.

[45] Hameed Hussain, Saif Ur Rehman Malik, Abdul Hameed, Samee Ullah Khan, Gage Bickler, Nasro Min-Allah, Muhammad Bilal Qureshi, Limin Zhang, Wang Yongji, Nasir Ghani, Joanna Kolodziej, Albert Y. Zomaya, Cheng-Zhong Xu, Pavan Balaji, Abhinav

Vishnu, Fredric Pinel, Johnatan E. Pecero, Dzmitry Kliazovich, Pascal Bouvry, Hongxiang Li, Lizhe Wang, Dan Chen, and Ammar Rayes. A survey on resource allocation in high performance distributed computing systems. *Parallel Computing*, 39(11):709–736, November 2013.

[46] Daulet Izbassarov and Metin Muradoglu. A front-tracking method for computational modeling of viscoelastic two-phase flow systems. *Journal of Non-Newtonian Fluid Mechanics*, 223:122–140, September 2015.

[47] Amin Ramiani Jafari. Phase change modelling with flexible source-based kinetics for non-equilibrium transitions. *International Journal of Thermal Sciences*, 159:106608, January 2021.

[48] H.W. Jia, P. Zhang, X. Fu, and S.C. Jiang. A numerical investigation of nucleate boiling at a constant surface temperature. *Applied Thermal Engineering*, 88:248–257, September 2015.

[49] Damir Juric and Grétar Tryggvason. Computations of boiling flows. *International Journal of Multiphase Flow*, 24(3):387–410, April 1998.

[50] Ankit Kalani and Satish G. Kandlikar. Pressure Drop Analysis Using the Homogeneous Model for Open Microchannel With Manifold (OMM). American Society of Mechanical Engineers Digital Collection, December 2014.

[51] Ankit Kalani and Satish G. Kandlikar. Flow patterns and heat transfer mechanisms during flow boiling over open microchannels in tapered manifold (OMM). *International Journal of Heat and Mass Transfer*, 89:494–504, October 2015.

[52] Ankit Kalani and Satish G. Kandlikar. A Photographic Study of Flow Boiling Stability on a Plain Surface With Tapered Manifold. American Society of Mechanical Engineers Digital Collection, November 2015.

[53] Y. Kamotani. Evaporator film coefficients of grooved heat pipes. In *3rd International Heat Pipe Conference*, Meeting Paper Archive. American Institute of Aeronautics and Astronautics, May 1978.

[54] Satish G. Kandlikar. High Flux Heat Removal with Microchannels—A Roadmap of Challenges and Opportunities. *Heat Transfer Engineering*, 26(8):5–14, October 2005. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/01457630591003655.

[55] Satish G. Kandlikar. Microscale to Macroscale—Extending Microscale Enhancement Techniques to Large-Scale Boiling Equipment. *Journal of Heat Transfer*, 144(5):050802, May 2022.

[56] Chirag R. Kharangate and Issam Mudawar. Review of computational studies on boiling and condensation. *International Journal of Heat and Mass Transfer*, 108:1164–1196, May 2017.

[57] Hyungdae Kim. Enhancement of critical heat flux in nucleate boiling of nanofluids: a state-of-art review. *Nanoscale Research Letters*, 6(1):1–18, 2011.

[58] Hyungdae Kim and Jacopo Buongiorno. Detection of liquid–vapor–solid triple contact line in two-phase heat transfer phenomena using high-speed infrared thermometry. *International Journal of Multiphase Flow*, 37(2):166–172, 2011.

[59] Rupak Kumar and Arup Kumar Das. Numerical study of boiling of liquid nitrogen at solid and liquid contact planes. *International Journal of Heat and Mass Transfer*, 183:122075, February 2022.

[60] Christian Kunkelmann, Khalid Ibrahem, Nils Schweizer, Stefan Herbert, Peter Stephan, and Tatiana Gambaryan-Roisman. The effect of three-phase contact line speed on local evaporative heat transfer: Experimental and numerical investigations. *International Journal of Heat and Mass Transfer*, 55(7):1896–1904, March 2012.

[61] Christian Kunkelmann and Peter Stephan. CFD Simulation of Boiling Flows Using the Volume-of-Fluid Method within OpenFOAM. *Numerical Heat Transfer, Part A: Applications*, 56(8):631–646, November 2009.

[62] Jeongmin Lee, Issam Mudawar, Mohammad M. Hasan, Henry K. Nahra, and Jeffrey R. Mackey. Experimental and computational investigation of flow boiling in microgravity. *International Journal of Heat and Mass Transfer*, 183:122237, February 2022.

[63] Woorim Lee and Gihun Son. Bubble Dynamics and Heat Transfer During Nucleate Boiling in a Microchannel. *Numerical Heat Transfer, Part A: Applications*, 53(10):1074–1090, January 2008.

[64] Woorim Lee, Gihun Son, and Jae Jun Jeong. Numerical Analysis of Bubble Growth and Departure from a Microcavity. *Numerical Heat Transfer, Part B: Fundamentals*, 58(5):323–342, November 2010.

[65] Jingwen Li, Zhen Yang, and Yuanyuan Duan. Numerical simulation of single bubble growth and heat transfer considering multi-parameter influence during nucleate pool boiling of water. *AIP Advances*, 11(12):125207, December 2021.

[66] Yongshi Liang, Cliff Y Guo, Xianglong Zhao, Qiang Qin, Yi Cheng, and Lixin He. CPFD simulation on particle behaviour in an entrained-flow gasifier. *Clean Energy*, 4(1):75–84, April 2020.

[67] J. H. Lienhard and V. K. Dhir. Hydrodynamic Prediction of Peak Pool-boiling Heat Fluxes from Finite Bodies. *Journal of Heat Transfer*, 95(2):152–158, May 1973.

[68] Hongli Liu, Jiguo Tang, Licheng Sun, Zhengyu Mo, and Guo Xie. An assessment and analysis of phase change models for the simulation of vapor bubble condensation. *International Journal of Heat and Mass Transfer*, 157:119924, August 2020.

[69] Qingming Liu and Björn Palm. Numerical study of bubbles rising and merging during convective boiling in micro-channels. *Applied Thermal Engineering*, 99:1141–1151, April 2016.

[70] Chanchal Loha, Himadri Chattopadhyay, and Pradip K. Chatterjee. Three dimensional kinetic modeling of fluidized bed biomass gasification. *Chemical Engineering Science*, 109:53–64, April 2014.

[71] J. López, J. Hernández, P. Gómez, and F. Faura. An improved PLIC-VOF method for tracking thin fluid structures in incompressible two-phase flows. *Journal of Computational Physics*, 208(1):51–74, September 2005.

[72] M. Magnini, B. Pulvirenti, and J. R. Thome. Numerical investigation of hydrodynamics and heat transfer of elongated bubbles during flow boiling in a microchannel. *International Journal of Heat and Mass Transfer*, 59:451–471, April 2013.

[73] Nenad Miljkovic, Ryan Enright, Youngsuk Nam, Ken Lopez, Nicholas Dou, Jean Sack, and Evelyn N Wang. Jumping-droplet-enhanced condensation on scalable superhydrophobic nanostructured surfaces. *Nano letters*, 13(1):179–187, 2012.

[74] Isaac Sir Newton. *Opticks: or, A treatise of the reflections, refractions, inflections & colours of light. Based on the 4th ed., London, 1730*. Number Book, Whole. Dover Publications, New York, 1952.

[75] William L Oberkampf, Timothy G Trucano, and Charles Hirsch. Verification, validation, and predictive capability in computational engineering and physics . *Applied Mechanics Reviews*, 57(5):345–384, 12 2004.

[76] Hajime Onishi, Takeaki Goto, Masashi Haruki, and Yukio Tada. Volume of fluid-based numerical analysis of a pump-driven phase change heat transport device. *International Journal of Heat and Mass Transfer*, 186:122429, May 2022.

[77] Stanley Osher and James A Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. *Journal of Computational Physics*, 79(1):12–49, November 1988.

[78] Jon P. Owejan, Jeffrey J. Gagliardo, Jacqueline M. Sergi, Satish G. Kandlikar, and Thomas A. Trabold. Water management studies in PEM fuel cells, Part I: Fuel cell design and in situ water distributions. *International Journal of Hydrogen Energy*, 34(8):3436–3444, May 2009.

[79] Furkan Özkan, Achim Wenka, Edgar Hansjosten, Peter Pfeifer, and Bettina Kraushaar-Czarnetzki. Numerical investigation of interfacial mass transfer in two phase flows using the VOF method. *Engineering Applications of Computational Fluid Mechanics*, 10(1):100–110, January 2016.

[80] Zhenhai Pan, Justin A. Weibel, and Suresh V. Garimella. A saturated-interface-volume phase change model for simulating flow boiling. *International Journal of Heat and Mass*

Transfer, 93:945–956, February 2016.

[81] Chinmay M. Patil and Satish G. Kandlikar. Review of the Manufacturing Techniques for Porous Surfaces Used in Enhanced Pool Boiling. *Heat Transfer Engineering*, 35(10):887–902, July 2014. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/01457632.2014.862141.

[82] I. Perez-Raya and S. G. Kandlikar. Chapter Three - Evaporation on a Planar Interface – Numerical Simulation and Theoretical Analysis of Heat and Mass Transport Processes. In Ephraim M. Sparrow, John P. Abraham, John M. Gorman, Thomas F. Irvine, and James P. Hartnett, editors, *Advances in Heat Transfer*, volume 48, pages 125–190. Elsevier, January 2016.

[83] Isaac Perez-Raya and Satish G. Kandlikar. Numerical modeling of interfacial heat and mass transport phenomena during a phase change using ANSYS-Fluent. *Numerical Heat Transfer, Part B: Fundamentals*, 70(4):322–339, October 2016.

[84] Isaac Perez-Raya and Satish G. Kandlikar. Discretization and implementation of a sharp interface model for interfacial heat and mass transfer during bubble growth. *International Journal of Heat and Mass Transfer*, 116:30–49, 2018.

[85] Isaac Perez-Raya and Satish G. Kandlikar. Numerical models to simulate heat and mass transfer at sharp interfaces in nucleate boiling. *Numerical Heat Transfer, Part A: Applications*, 74(10):1583–1610, November 2018.

[86] Paul (Pinghua) Zhao, Peter J. O'Rourke, and Dale Snider. Three-dimensional simulation of liquid injection, film formation and transport, in fluidized beds. *Particuology*, 7(5):337–346, October 2009.

[87] Stéphane Popinet and Stéphane Zaleski. A front-tracking algorithm for accurate representation of surface tension. *International Journal for Numerical Methods in Fluids*, 30(6):775–793, 1999. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/%28SICI%291097-0363%2819990730%2930%3A6%3C775%3A%3AAID-FLD864%3E3.0.CO%3B2-%23.

[88] R. Raj, C. Kunkelmann, P. Stephan, J. Plawsky, and J. Kim. Contact line behavior for a highly wetting fluid under superheated conditions. *International Journal of Heat and Mass Transfer*, 55(9):2664–2675, April 2012.

[89] Harshal S. Raut, Amitabh Bhattacharya, and Atul Sharma. Sustaining nucleate boiling in zero gravity using asymmetric sinusoidal base-plate oscillation. *International Journal of Heat and Mass Transfer*, 184:122262, March 2022.

[90] Patrick J Roache. *Verification and validation in computational science and engineering*, volume 895. Hermosa Albuquerque, NM, 1998.

[91] Patrick J Roache. Verification of codes and calculations. *AIAA journal*, 36(5):696–702, 1998.

[92] Rochester Institute of Technology. Research Computing Services, 2022.

[93] J. W. Rose. Dropwise condensation theory and experiment: a review. *Proceedings of the Institution of Mechanical Engineers Part a-Journal of Power and Energy*, 216(A2):115–128, 2002.

[94] Avijit Saha, Temistocle Grenga, Abhishek Y. Deshmukh, Jörn Hinrichs, Mathis Bode, and Heinz Pitsch. Numerical modeling of single droplet flash boiling behavior of e-fuels considering internal and external vaporization. *Fuel*, 308:121934, January 2022.

[95] Nastaran A. Samani, Rajan K. Thapa, Britt M. E. Moldestad, and Marianne S. Eikeland. Evaluating the impacts of temperature on a bubbling fluidized bed biomass gasification using CPFD simulation model. *IFAC-PapersOnLine*, 55(20):618–623, January 2022.

[96] Yohei Sato and Bojan Ničeno. A conservative local interface sharpening scheme for the constrained interpolation profile method. *International Journal for Numerical Methods in Fluids*, 70(4):441–467, 2012.

[97] Yohei Sato and Bojan Ničeno. A sharp-interface phase change model for a mass-conservative interface tracking method. *Journal of Computational Physics*, 249:127–161, September 2013.

[98] Yohei Sato and Bojan Niceno. Nucleate pool boiling simulations using the interface tracking method: Boiling regime from discrete bubble to vapor mushroom region. *International Journal of Heat and Mass Transfer*, 105:505–524, February 2017.

[99] Yohei Sato and Bojan Niceno. Pool boiling simulation using an interface tracking method: From nucleate boiling to film boiling regime through critical heat flux. *International Journal of Heat and Mass Transfer*, 125:876–890, October 2018.

[100] Ruben Scardovelli and Stéphane Zaleski. Direct Numerical Simulation of Free-Surface and Interfacial Flow. *Annual Review of Fluid Mechanics*, 31(1):567–603, 1999.

[101] L. E. Scriven. On the dynamics of phase growth. *Chemical Engineering Science*, 10(1):1–13, April 1959.

[102] Xiaopeng Shang, Xuan Zhang, Thien-Binh Nguyen, and Tuan Tran. Direct numerical simulation of evaporating droplets based on a sharp-interface algebraic VOF approach. *International Journal of Heat and Mass Transfer*, 184:122282, March 2022.

[103] ATUL SHARMA. Level set method for computational multi-fluid dynamics: A review on developments, applications and analysis. *Sadhana*, 40(3):627–652, May 2015.

[104] Robert R Sharp. *The nature of liquid film evaporation during nucleate boiling*. National Aeronautics and Space Administration, 1964.

[105] Simon Peter Shipkowski, Isaac Perez-Raya, and Satish G. Kandlikar. A Proposed Approach for Accurate Estimation of Interface Surface Area in Multiphase Simulations. American Society of Mechanical Engineers Digital Collection, October 2020.

[106] S.P. Shipkowski and I Perez-Raya. Precise and analytical calculation of interface surface area in sharp interfaces and multiphase modeling. *International Journal of Heat and Mass Transfer*, 202:123683, March 2023.

[107] Wei Shyy, HS Udaykumar, Madhukar M Rao, and Richard W Smith. *Computational fluid dynamics with moving boundaries*. Courier Corporation, 2012.

[108] R. Siegel and E. G. Keshock. Effects of reduced gravity on nucleate boiling bubble dynamics in saturated water. *AIChE Journal*, 10(4):509–517, 1964.

[109] D. M. Snider. An Incompressible Three-Dimensional Multiphase Particle-in-Cell Model for Dense Particle Flows. *Journal of Computational Physics*, 170(2):523–549, July 2001.

[110] ANSYS Inc. Software. ANSYS-Fluent 12.0/12.1 Documentation Web Site, 2020.

[111] G. Son, V. K. Dhir, and N. Ramanujapu. Dynamics and Heat Transfer Associated With a Single Bubble During Nucleate Boiling on a Horizontal Surface. *Journal of Heat Transfer*, 121(3):623–631, August 1999.

[112] Gihun Son. A Numerical Method for Bubble Motion with Phase Change. *Numerical Heat Transfer, Part B: Fundamentals*, 39(5):509–523, May 2001.

[113] Gihun Son. Efficient Implementation of a Coupled Level-Set and Volume-of-Fluid Method for Three-Dimensional Incompressible Two-Phase Flows. *Numerical Heat Transfer, Part B: Fundamentals*, 43(6):549–565, June 2003.

[114] Gihun Son and Vijay K. Dhir. Numerical simulation of nucleate boiling on a horizontal surface at high heat fluxes. *International Journal of Heat and Mass Transfer*, 51(9):2566–2582, May 2008.

[115] Gihun Son and Vijay K. Dhir. Three-dimensional simulation of saturated film boiling on a horizontal cylinder. *International Journal of Heat and Mass Transfer*, 51(5):1156–1167, March 2008.

[116] Mark E. Steinke and Satish G. Kandlikar. Single-phase liquid friction factors in microchannels. *International Journal of Thermal Sciences*, 45(11):1073–1083, November 2006.

[117] Mark Sussman and Elbridge Gerry Puckett. A Coupled Level Set and Volume-of-Fluid Method for Computing 3D and Axisymmetric Incompressible Two-Phase Flows. *Journal of Computational Physics*, 162(2):301–337, August 2000.

[118] Sébastien Tanguy, Michaël Sagan, Benjamin Lalanne, Frédéric Couderc, and Catherine Colin. Benchmarks and numerical methods for the simulation of boiling flows. *Journal of Computational Physics*, 264:1–22, May 2014.

[119] T. Theofanous. The boiling crisis phenomenon Part I: nucleation and nucleate boiling heat transfer. *Experimental Thermal and Fluid Science*, 26(6-7):775–792, 2002.

[120] T. G. Theofanous, T. N. Dinh, J. P. Tu, and A. T. Dinh. The boiling crisis phenomenon - Part II: dryout dynamics and burnout. *Experimental Thermal and Fluid Science*, 26(6-7):793–810, August 2002.

[121] Yijie Tong, Hoseong Lee, Woobin Kang, and Honghyun Cho. Energy and exergy comparison of a flat-plate solar collector using water, Al2O3 nanofluid, and CuO nanofluid. *Applied Thermal Engineering*, 159:113959, August 2019.

[122] Yeng-Yung Tsui, Cheng-Yen Liu, and Shi-Wen Lin. Coupled level-set and volume-of-fluid method for two-phase flow calculations. *Numerical Heat Transfer, Part B: Fundamentals*, 71(2):173–185, February 2017.

[123] D.B. Tuckerman and R.F.W. Pease. High-performance heat sinking for VLSI. *IEEE Electron Device Letters*, 2(5):126–129, May 1981. Conference Name: IEEE Electron Device Letters.

[124] M. van Sint Annaland, W. Dijkhuizen, N. G. Deen, and J. a. M. Kuipers. Numerical simulation of behavior of gas bubbles using a 3-D front-tracking method. *AIChE Journal*, 52(1):99–110, 2006. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/aic.10607.

[125] Sjoerd Van Stralen and Robert Cole. *Boiling phenomena: physicochemical and engineering fundamentals and applications*, volume 2. Hemisphere, 1979.

[126] Henk Kaarle Versteeg and Weeratunge Malalasekera. *An introduction to computational fluid dynamics: the finite volume method*. Pearson education, 2007.

[127] P. C. Wayner. Adsorption and capillary condensation at the contact line in change of phase heat transfer. *International Journal of Heat and Mass Transfer*, 25(5):707–713, May 1982.

[128] P. C. Wayner, Y. K. Kao, and L. V. LaCroix. The interline heat-transfer coefficient of an evaporating wetting film. *International Journal of Heat and Mass Transfer*, 19(5):487–492, May 1976.

[129] Peter C Wayner. The interfacial profile in the contact line region and the Young—Dupré equation. *Journal of Colloid and Interface Science*, 88(1):294–295, July 1982.

[130] Samuel W. J. Welch and Thami Rachidi. Numerical Computation of Film Boiling Including Conjugate Heat Transfer. *Numerical Heat Transfer, Part B: Fundamentals*, 42(1):35–53, July 2002.

[131] Samuel W. J. Welch and John Wilson. A Volume of Fluid Based Method for Fluid Flows with Phase Change. *Journal of Computational Physics*, 160(2):662–682, May 2000.

[132] H. L. Wen, C. H. Yu, and Tony W. H. Sheu. On the development of LS-assisted VOF method for incompressible interfacial flows. *Journal of Computational Physics*, 406:109188, April 2020.

[133] Jinfeng Wu and Vijay K. Dhir. Numerical Simulation of Dynamics and Heat Transfer Associated With a Single Bubble in Subcooled Boiling and in the Presence of Noncondensables. *Journal of Heat Transfer*, 133(4), January 2011.

[134] Xue Xiaodi, Zheng Hongfei, He Kaiyan, Chen Zhili, Tao Tao, and Xie Guo. Experimental study on a new solar boiling water system with holistic track solar funnel concentrator. *Energy*, 35(2):692–697, February 2010.

[135] Yan Yang, Liang-ming Pan, and Jian-jun Xu. Effects of microgravity on Marangoni convection and growth characteristic of a single bubble. *Acta Astronautica*, 100:129–139, July 2014.

[136] Miad Yazdani, Thomas Radcliff, Marios Soteriou, and Abbas A. Alahyari. A high-fidelity approach towards simulation of pool boiling. *Physics of Fluids*, 28(1):012111, January 2016.

[137] Ilhwan Yeo and Seunghyun Lee. 2D computational investigation into transport phenomena of subcooled and saturated flow boiling in large length to diameter ratio microchannel heat sinks. *International Journal of Heat and Mass Transfer*, 183:122128, February 2022.

[138] Tianhao Yi, Xu Chu, Bin Wang, Jingyi Wu, and Guang Yang. Numerical simulation of single bubble evolution in low gravity with fluctuation. *International Communications in Heat and Mass Transfer*, 130:105828, January 2022.

[139] L Youngs D. Time-Dependent Multi-Material Flow with Large Fluid Distortion. *Numerical Methods for Fluid Dynamics*, 1982. Publisher: Academic Press.

[140] Q. Zhao and B. M. Burnside. Dropwise condensation of steam on ion implanted condenser surfaces. *Heat Recovery Systems and CHP*, 14(5):525–534, September 1994.

[141] Shuo Zhao, Jie Zhang, and Ming-Jiu Ni. Boiling and evaporation model for liquid-gas flows: A sharp and conservative method based on the geometrical VOF approach. *Journal of Computational Physics*, 452:110908, March 2022.

[142] Shao-Fei Zheng, Ulrich Gross, and Xiao-Dong Wang. Dropwise condensation: From fundamentals of wetting, nucleation, and droplet mobility to performance improvement by advanced functional surfaces. *Advances in Colloid and Interface Science*, 295:102503, September 2021.

Appendices

Appendix A

Computational Appendix

This is an appendix of simulation/computational details/considerations, programming code, and pseudo code referenced elsewhere in this document.

A.1. PLIC-ASB Implementation Pseudo Code

Below is generic C-like code to facilitate implementation of the Piece-wise Linear Interface Calculation Method by interested parties. This pseudo code is provided for more universal applicability for a variety of programming languages, CFD software, simulation suites, etc.

Here n and m are the indices of a grid of square cells in a simulation using axial-symmetric geometry with "m" counting away from the axis of rotational symmetry. N and M are the re-

spective maximum values of n and m with zero the minimums of each. Vf is the volume fraction of the cell with indices (n,m) and "NormVector[]" is the normal vector obtained explicitly via the "Gradient()" function, a user's personal scheme, or through included functions of the user's software. The variables nx and ny are the x and y components of the normal vector, respectively, while tx and ty are the transformed components of the normal vector, respectively.

'InterfacePLICASB(Vf, NormVector []),{N,M}' is the primary function whose arguments here are the volume fraction of the cell of interest and the normal vector found based on the volume fraction distribution about that cell.

The N and M refer to the total domain of the computational simulation.

Please see the first subsection (A.1.1) for an illustrative 2-dimensional implementation case and the second subsection (A.1.2) for an axial-symmetric implementation case.

A.1.1. PLIC-ASB Implementation: 2D Pseudo Code Example

```
InterfacePLICASB( Vf, NormVector[] ),{N,M}
/// The m and n loops are usually part
/// of the existing code...
////////////////////////////////////////
```

```
/// Start an 'm' loop
n=0, m=0;

While Loop Over m from 0 to M ( Cell[n,,m] )
{
/// Start \n" loop, Identify and select interface
/// cells. The inclusive condition is just that
/// the cell contains multiple phases.
/// In computational terms an upper and lower
/// bound is dependant on the users conditions
/// but a sufficiently close to 0 or 1 value may
/// best be considered non-fractional. The degree
/// to which this is a matter of approximation,
/// precision, or smooth operating procedure should
/// be obvious.
    While Loop Over n from 0 to N ( Cell[n,m] )
    {

    /// 'If conditions,' -> write in the negative for best
    /// practices     ~If(Vf > 0.001 &&
    ///            Vf < 0.999 && Any Other User Conditions)
```

```
If(Vf < 0.001 OR Vf > 0.999 OR =/= Any Other User Conditions)
    {
    /// option to return: 'not interface cell'
    /// or lint = 0;
    n = n+1 ;
    /// loop moves to next cell
    /// if this is not an interface cell

    }
    else
    {
    /// Find x and y gradient components
    NormVector = GradientOperator(Cell n, m );

    Vector[x] = DotProduct of ( x-hat, NormVector) ;
    Vector[y] = DotProduct of ( y-hat, NormVector) ;

    /// The following two lines are merely descriptive
    nx = Vector[x];
    ny = Vector[y];
```

```
/// Apply an arbitrary in plane orthogonal rotation
/// operator to the vector (components) to form a
/// vector that is parallel with the interface
tx = (-1) * Vector[y];
ty = Vector[x];

/// Convert volume fraction 'Vf' to symmetric form:
If (Vf > 0.5,
    {F = 1- Vf }
 Else
    {F = Vf }

/// Calculate *** F independent form \int" ***
int1 = dx/cos[theta];

/// Calculate *** F dependent form \int" ***
int2 = Sqrt[2*F*dx*dx / ( Sin[theta]*cos[theta] ) ]

/// 'int' Anzats: int = int2, but if int2 > int1,
/// int1 must be int: int = int1
```

```
          If(int1 > int2)
                 {
                 Lint = int2;
                 }
                 Else
                 {
                 Lint = int1;
                 }
         /// loop continues over 'n'
    n=n+1;

    }  /// Close 'n' loop bracket

/// reset 'n' and advance 'm'

n = 0;
m = m+1;

}
/// close 'm' loop bracket
///  Total Calculation complete for all n & m
```

A.1.2. PLIC-ASB Implementation: Axial Symmetric Example

```
InterfacePLICASB( Vf, NormVector[] ),{N,M}
/// The m and n loops are usually part
/// of the existing code...
//////////////////////////////////////////


/// Start the 'm' loop


n=0, m=0;


While Loop Over m from 0 to M ( Cell[n,,m] )
{


/// Start \n" loop, Identify and select interface
/// cells. The inclusive condition is just that
/// the cell contains multiple phases.
/// In computational terms an upper and lower
/// bound is dependant on the users conditions
```

```
/// but a sufficiently close to 0 or 1 value may
/// best be considered non-fractional. The degree
/// to which this is a matter of approximation,
/// precision, or smooth operating procedure should
/// be obvious.

    While Loop Over n from 0 to N ( Cell[n,m] )
    {

    /// 'If conditions,' -> write in the negative for best
    /// practices     ~If(Vf > 0.001 &&
    ///            Vf < 0.999 && Any Other User Conditions)

    If(Vf < 0.001 OR Vf > 0.999 OR =/= Any Other User Conditions)
        {
        /// option to return: 'not interface cell'
        /// or lint = 0;

          n = n+1 ;
        /// loop moves to next cell
        /// if this is not an interface cell
```

```
}
else
{

/// Find x and y gradient components
NormVector = GradientOperator(Cell n, m );

Vector[x] = DotProduct of ( x-hat, NormVector) ;
Vector[y] = DotProduct of ( y-hat, NormVector) ;

/// The following two lines are merely descriptive
nx = Vector[x];
ny = Vector[y];

/// Apply an arbitrary in plane orthogonal rotation
/// operator to the vector (components) to form a
/// vector that is parallel with the interface
tx = (-1) * Vector[y];
ty = Vector[x];

/// Convert volume fraction 'Vf' to symmetric form:
```

```
 If (Vf > 0.5,
    {F = 1- Vf }
 Else
    {F = Vf }

/// Calculate *** F independent form \int" ***
int1 = dx/cos[theta];

/// Calculate *** F dependent form \int" ***
int2 = Sqrt[2*F*dx*dx / ( Sin[theta]*cos[theta] ) ]

/// 'int' Anzats: int = int2, but if int2 > int1,
/// int1 must be int: int = int1
    If(int1 > int2)
        {
        Lint = int2;
        }
        Else
        {
        Lint = int1;
        }
```

```
        /// Axial-symmetric area calculation
        /// (greater accuracy can be added,
        /// contingent on user methods)
        R = dx * ( m + 1/2 );

        Ring = 2 * Pi * R;
        }

        /// *** This is the result of the
        /// interfacePLICASB() function ***

int(n,m)=[Lint * Ring ];
Total_int = Total_int + int(n,m);

/// loop continues over 'n'
n=n+1;

    }  /// Close 'n' loop bracket

/// reset 'n' and advance 'm'
```

```
n = 0;

m = m+1;


} /// close 'm' loop bracket

///  Total Calculation complete for all n & m
```