2009

A Volume-of-Fluid Based Numerical Simulation of Solidification in Binary Alloys on Fixed Non-uniform Co-located Grids

Mehdi Farrokhnejad Roudsari

A Volume-of-Fluid Based Numerical Simulation of Solidification in Binary Alloys on Fixed Non-uniform Co-located Grids

(Spine title: Numerical Simulation of Solidification in Binary Alloys)
(Thesis format: Monograph)

by

Mehdi Farrokhnejad Roudsari

Graduate Program in Engineering
Department of Mechanical & Material Engineering

Submitted in Partial fulfillment
Of the requirements for the degree of
Master of Engineering Science

The School of Graduate and Postdoctoral Studies
The University of Western Ontario
London, Ontario, Canada
April, 2009

Abstract

In this work, the author presents a platform for the modeling of mold filling and solidification of binary alloys with properties similar to Mg alloys. A volume-of-fluid (VOF) based method to capture the interface between solid and liquid in a solidification process on a fixed 2D non-uniform grid, developed for implementation in a co-located finite volume framework, is presented. Contrary to other works, to update the volume fraction of fluid in the field, a link between source-based type of energy equation and VOF algorithm is described and implemented. A new approximation to the pressure gradient is presented to remove all "Spurious Currents" [1] resulting from pressure jumps in the vicinity of the interface.

Based upon the work presented, it is summarized that the present combination of the equations are not only computationally simple to implement and upgrade to a 3D problem, but also provides an excellent platform to capture the interface between constituents in a die-casting process including solidification and mold filling process. This will lead to a better understanding of the die-casting process.

Keywords: Mg Solidification modelling, binary alloy solidification modelling, Solidification, Volume of fluid, VOF, Die-casting modelling, Heat Transfer, incompressible flow, finite volume method

Acknowledgment

I would like to express my deepest gratitude to my supervisors, Profs. A.G. Straatman and J.T. Wood, for the guidance provided throughout my research, their constructive criticism, many valuable suggestions and continuous support.

I also would like to thank Meridian technologies Inc. for their contribution in financial support.

I would like to thank all my current and previous colleagues of the Prof. Straatman's Advanced Fluid Dynamic Lab with whom I had numerous productive discussions, in particular Dr. S.M. Karimian, Dr F. Bahramian, Mr. L. Betchen, Mr. K. Sultan, Mr C. Degroot and J. Cepek.

I also owe many thanks to Prof. G. Biswas, visiting professor from the Indian Institute of Technology, Kanpour, for all the helpful discussions in regard with VOF method.

My thanks also go to AUTO21 Network of Centers of Excellence, an automotive research and development program that focuses on issues relating to the automobile industry in 21st century, for providing all educational and financial support in this project.

Finally, I take the opportunity to thank my family and friends for their support and interest throughout this research. I am quite sure; I could not have brought this work to this degree of completion without their sustained encouragement.

Table of Contents

List of Figures

List of Tables

Nomenclature

A	Small number	
a, b	Coefficient in the discretization equation	
C, M	Capacitance matrices	
$F(T)$	Liquid-fraction temperature relationship	
f	Volume fraction	
f_l	Volume fraction of liquid	
H	Total enthalpy	
K	Stiffness matrix	
k	Thermal conductivity	$[W/mK]$
k_p	Partition coefficient	
L_{fs}	Latent heat of solidification	$[J/kg]$
L	Distance from the cell center	[m]
$\vec{n}$	Normal to the interface	
P	Pressure	[pa]
S	Source term	
S_t	Source term in Energy equation	
S_u, S_v	Source terms in momentum equations	
T	Temperature	[K]
t	Time	[s]
Δt	Time step	
T_m	Melting temperature	[K]
T_l or T_L	Liquidus temperature	[K]
T_E	Eutectic temperature	[k]
$\vec{\hat{u}}_e$	Advecting velocity	$[m/s]$
$\vec{u}$	Field velocity	$[m/s]$
$\vec{u}_s$	Solid velocity	$[m/s]$
$\vec{u}_l$	Liquid velocity	$[m/s]$
V	Volume	$[m^3]$

Greek Symbols

ρ	Density	$[kg/m^3]$
μ	Dynamic viscosity	$[kg/ms]$
∇	Gradient operator	
κ	Permeability coefficient	
ε	Porosity	
β	Volumetric Thermal Expansion	[1/K]

Subscripts

b	Body force

l	Liquid/liquidus
m	Melting
mix	mixture
nb	Neighbor points
P	Node point P
ref	reference
s	solid
v	vertex

Superscripts

m	Iteration level
old	Previous time step value

Chapter 1

1. Introduction

1.1- Motivation of studying the solidification of Mg Die-Casting

The past years have witnessed a significant growth in utilization of magnesium (Mg) alloys in the automotive industry. Due to the excellent fluidity of molten Mg alloys and low specific weight of Mg, along with their excellent machinability and a good recycling potential, this group of alloys is a good candidate not only to be used in automobile industry in order to reduce the total weight of vehicles and meet stringent fuel economy standards, but also be suitable for the manufacture of other thin section die castings. These capabilities increase the global interest of using Mg more and more [2, 3].

Although auto industries have been using Mg alloys dramatically due to their excellent potential, they are still not used to the same extent as competing material such as aluminum and plastics due to two main barriers [2, 3]:

1. The increased cost relative to aluminum and steel
2. The inability to control as-cast microstructure to the level required for optimized mechanical properties.

Industrial products are directly influenced by the mechanical properties of the material from which they are processed, and furthermore, the mechanical properties of the materials are closely associated with their solidification microstructure. Mg die-casting, due to the extreme flow and solidification rates inherent in the process, makes equilibrium solidification theories inapplicable. Furthermore, the relationship between

casting parameters, microstructure and mechanical properties of die-casting Mg alloys are not well understood [2, 3].

Most of the previous work related to mechanical properties in Mg die-casting has focused on the casting of test bars under controlled Lab conditions. The results are predictable and have consistent mechanical properties [4]. However, for more complex components, the dynamic nature of the die-casting process combined with the complex geometries of castings may cause variation in the filling pattern, cooling rate and feeding conditions during solidification. In other words, some characteristic features may exist in the microstructure of this die casting that may not be present in cast test bars [4].

The need to achieve consistent and predictable properties is becoming increasingly important as applications move to components that require a higher level of structural integrity [4]. Therefore, a better understanding of the development of microstructure during the die-casting process and the correlation between mechanical properties and microstructure are required to ensure high quality for die cast Mg components.

The present work is a small part of a larger project that focuses on techniques of die-casting and the use of numerical simulation to improve the die-casting process. According to the flow chart shown in Figure 1-1, the ultimate goal is to develop tools to predict local mechanical properties of a thin-walled magnesium die-casting based on numerical simulation.

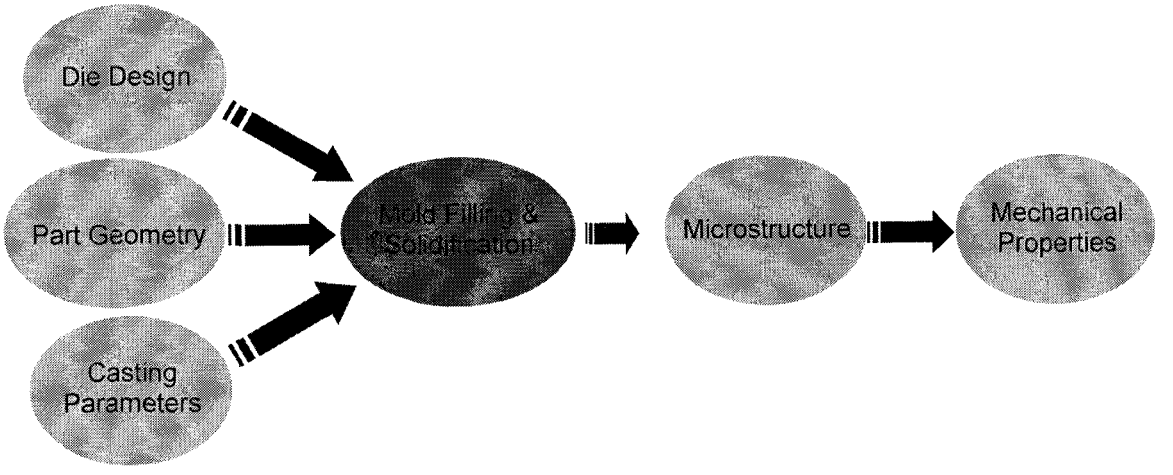

Figure 1-1, Flow chart of necessary sections to achieve ultimate goal

The aim of this research, as a small part of the whole project, is to provide a platform for future modeling of mold filling processes as well as being able to simulate a simplified solidification process of binary alloys, preferably, those which have properties similar to Mg alloys, at the same time.

In the following section, phase change phenomenon will be explained in the most relevant way to the current work to give the reader an idea about the crucial issues related to the modeling of solidification processes.

1.2- An introduction to phase- change problems

Phase change refers to the change of state, vapor–liquid or liquid-solid or solid-vapor, which a substance undergoes in response to an energy extraction/input. The most relevant example is the liquid-solid transition or solidification that happens during the casting of a multi-component metal alloys. Significant work has been done on the solidification of metal alloys in different specific areas; however, a critical aspect in much of this work is the development and application of computational heat and mass

transfer models. Having this in mind, the objective of this section is to outline some theoretical groundwork required for the development of metal alloy solidification models.

1.2.1- Requirement of Numerical simulation

In early years, analytical methods were the only way available to make mathematically an understanding of physical processes involving phase change phenomena. Although analytical methods offer an exact solution and are mathematically compact, due to their limitations, they are mainly only suitable for one-dimensional cases of infinite or semi-infinite regions with simple initial and boundary conditions and constant thermal properties [5].

In practice, solidification or any phase change problems are hardly ever one-dimensional; initial and boundary conditions are usually complex, and the variation of thermo-physical properties are significant [6]. Mathematical modeling and computer simulation have become more popular and the most economical and fastest way to provide a broad understanding of the practical processes involving phase change with the rise of high speed digital computers [6].

1.2.2- Solidification Process

Numerical modeling of phase change problems, particularly, the solidification of alloys is very challenging. A perfect model requires consideration of the heat and mass transfer phenomena that happen across a wide range of time and length scales. For example, the time scale can vary from $10^{-4} - 10^{2}$ (s) associated with nucleation kinetics and an industrial solidification process, respectively. The length scale can also range from submicron, associated with solid-liquid interface, to industrial process scale with a length

scale of meters [7]. Understanding how the process of solidification happens may give individuals the motivation and insight to remove modeling difficulties.

1.2.2.1- Nucleation

The crystallization of a large amount of material from a single point of nucleation results in a single crystal. In engineering materials, single crystals are produced only under carefully controlled conditions. The expense of producing single crystal materials is only necessary for special applications, such as turbine engine blades, solar cells, and piezoelectric materials. Normally, when a material begins to solidify, multiple crystals begin to grow in the liquid and a polycrystalline (more than one crystal) solid forms.

The moment a crystal begins to grow is known as nucleation and the point where it occurs is the nucleation point. At the solidification temperature, atoms of a liquid, such as melted metal, begin to bond together at the nucleation points and start to form crystals. The final sizes of the individual crystals depend on the number of nucleation points. The crystals increase in size by the progressive addition of atoms and grow until they impinge upon adjacent growing crystals [8]. See Figure 1-2.

A crystal is usually referred to as a grain in engineering materials. A grain is simply a crystal without smooth faces because its growth is impeded by contact with another grain or a boundary surface. The interface formed between grains is called a grain boundary. The atoms between the grains (at the grain boundaries) have no crystalline structure and are said to be disordered [8].

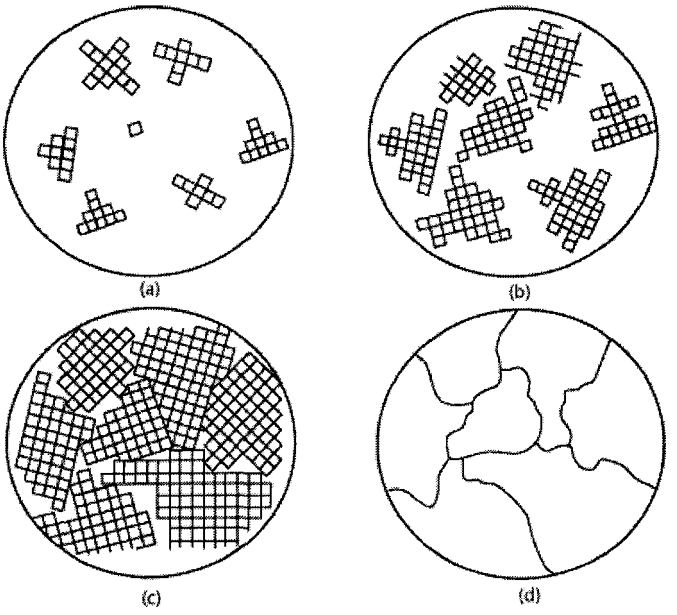

Figure 1-2, a) Nucleation of crystals b) crystal growth c) irregular grains form as crystals grow together d) grain boundaries as seen in a microscope [8]

Grains are usually large enough to be visible under an ordinary light microscope or even to the naked eye. The spangles that are seen on newly galvanized metals are grains. Rapid cooling generally results in more nucleation points and smaller grains (a fine grain structure). Slow cooling generally results in larger grains which will have lower strength, hardness and ductility [8].

1.2.2.2- Dendrites

The shape and size of crystallized grains depend on the conditions of their growth during solidification, mainly on the rate and direction of heat removal, the temperature of molten metal and the concentration of impurities [8]. In most metals, the crystals that form in the liquid during freezing generally follow a pattern consisting of a main branch with many appendages. A crystal with this morphology slightly resembles a pine tree and is called a dendrite, which means branching. It has been established that crystals grow

with the highest rate along the planes and directions where the atoms are packed most closely. Thus, long branches grow first, which are called the first-order dendritic axes. Then secondary dendrite arms branch off the primary arm, and tertiary arms off the secondary arms, etc. [8].Figure 1-3 shows how a cubic crystal can grow in a melt in three dimensions, which correspond to the six faces of the cube. For clarity of illustration, the adding of unit cells with continued solidification from the six faces is shown simply as lines.

During solidification of a polycrystalline material, many dendritic crystals form and grow until they eventually become large enough to impinge upon each other. Eventually, the inter-dendritic spaces between the dendrite arms crystallize to yield a more regular crystal. If there is not enough liquid material to fill these spaces, some crystals may retain the dendritic shape. The original dendritic pattern may not be apparent when examining the microstructure of a material. However, dendrites can often be seen in solidification voids that sometimes occur in castings or welds [8].

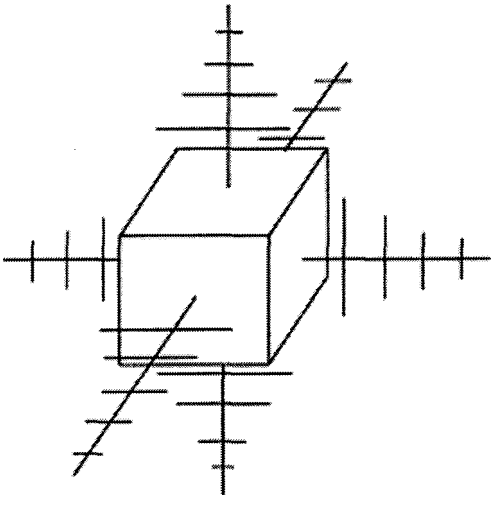

Figure 1-3, Dendrite growth direction [8]

1.2.2.3- Micro-Macrosegregation

When two constituents of an alloy have the same type of crystal lattice and their atomic diameters differ but slightly from each other, they can be mutually soluble in the solid state [8].

Consider a constitutional diagram for alloys with complete mutual solubility of constituent in the liquid and solid states. Figure 1-4.

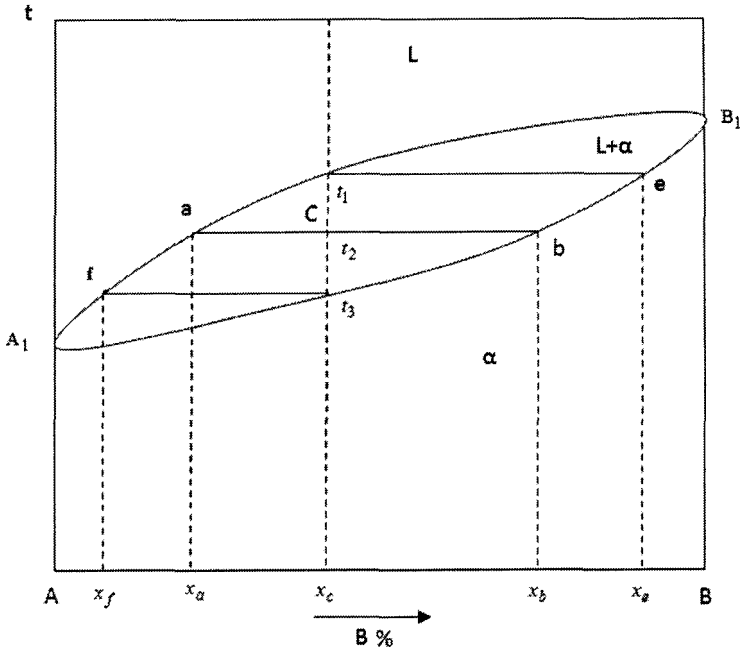

Figure 1-4, Constitutional diagram for alloys with complete solubility of constituents in the liquid and solid states

A diagram of this type should consist of the liquidus and solidus lines, which intersect at the points of solidification of pure constituents A and B. Now consider the solidification of an alloy of this system with composition x_c. As temperature drops to t_1, solidification begins in the liquid solution. Although it might be supposed that the crystals frozen out of liquid solution have the same composition as the liquid, actually,

the crystals freezing out of the liquid solution of the composition x_c at temperature t_1 are strongly enriched with the constituent having a higher melting point, i.e. with B. With further cooling to temperature t_2 the liquid of composition x_a, i.e. enriched with constituent A, will be in equilibrium with the solid solution of composition x_b. During slow cooling, the constituent A diffuses from the liquid into crystals of composition x_e, and this changes the composition to x_b. At the end of the solidification at temperature t_3, all crystals will have the same composition x_c [8].

After solidification, a single-phase structure is observed in an alloy. If the solidification occurs with fast cooling condition, as is usually the case in practical applications, there is no time for diffusion equalization of the composition of crystals freezing out at the temperature above t_3. As the result the composition may vary from crystal to crystal and in an individual crystal itself. Inner portion or layer of a crystal will be richer in constituent A. This phenomenon of inhomogeneous composition in crystals is called microsegregation. Crystals which have solidified first at the surface of for example an ingot will be rich in constituent B, while the last-to-solidify crystals in inner part of the ingot will be rich in constituent A. The result is what is called macrosegregation [8].

Segregation usually plays a negative part, particularly, when harmful impurities are unevenly distributed in the metal. A high concentration of impurities can cause premature failure of machine elements they are made of [8].

1.2.2.4- Industrial Solidification Process

An industrial solidification process can be simplified as the casting of a molten alloy in a rectangular mold. By setting the temperature of walls to below the equilibrium

liquidus temperature of the alloy, a solid layer forms on the mold and grows into the molten alloy. At the equilibrium liquidus temperature a sharp interface will exist between the solid and liquid phase if the growth of the solid layer is perfectly controlled.

Due to the kinetics, curvature and compositional effects, the temperature of liquid at the interface can fall below the liquidus temperature -under-cooling condition- causing an interface perturbation of the solid into the liquid, and creating a suitable region for growth. This kind of unstable solid growth behavior leads to a breakdown in the solid-liquid interface resulting formation of a "mushy" region consisting of a mixture of solid and liquid [7]. See Figure.1-5

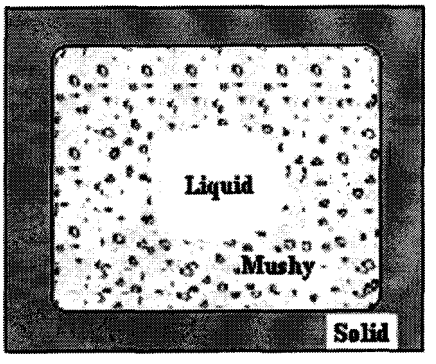

Figure 1-5, Schematic of a solidification system

1.2.3- Issues in modelling of a solidification Process

1.2.3.1- Mushy Region Modeling

One of the central issues in the modeling of alloy systems is the treatment of the mushy region. The solid fractions in the mushy region can have different morphologies base on the condition of heat removal during solidification. A sharp interface between solid-liquid can be recognized only at the microscopic length scale. What exactly happens in the mushy region is; as explained in section 1.2.2, at very early stages of

solidification, the solid parts in the mushy region form a fixed arrangement of columnar crystals attached to the fully solid layer on the mold wall. The reason behind the columnar growth is that crystals grow preferably in the direction opposite to those of heat removal. And for this reason, oriented heat removal results in the formation of stretched (columnar) crystals. In most general cases, after the columnar growth phase, free crystals, so called equiaxed crystals, form in the mushy region. Equiaxed crystals are formed if heat is removed from a growing crystal roughly with the same intensity in all directions. These types of crystals are not attached to any solid layer and can be moved by the motion of the surrounding molten alloy [7, 9]. See Figure 1-6.

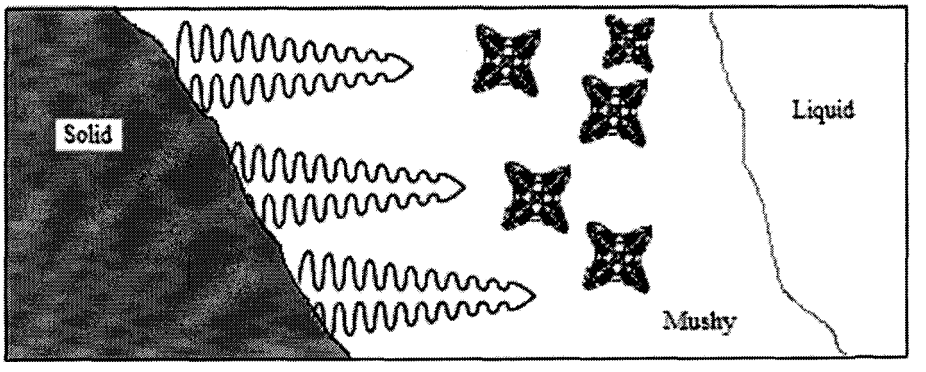

Figure 1-6, Schematic of the phenomena in mushy region

Based on the type of the material, there are different ways to look at this region. In a general system with randomly equiaxed nucleated grains, the solid is located throughout the liquid. In this case both liquid and solid phases have distinct velocities. (See Figure1-7)

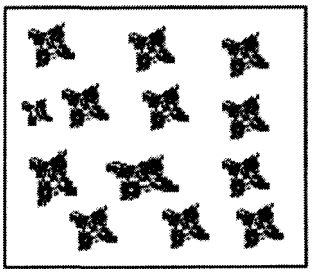

Figure 1-7, Schematic of a general equiaxed mushy region

The analysis of such a system requires a two-phase modeling approach [10]. Voller et al. [11], however, noted that there are two limiting cases under which the general model shown in Figure 1-7 can be reduced to situations for which a one-phase model can be developed based on mixture properties.

Case1: "A mushy fluid". In this case, it is assumed that the solid phase is fully dispersed in the liquid phase and both phases velocities are equal [11]. See Figure 1-8. Waxy materials have shown a reasonable compatibility with this case [10].

Case2: "A columnar zone". In this case, the solid phase has a dendrite shape and its matrix is distinct from the liquid. The solid matrix is either fixed or moves with a prescribed velocity. Reasonably compatible cases occur in static or continuous casting of alloys [10]. See figure 1-9.

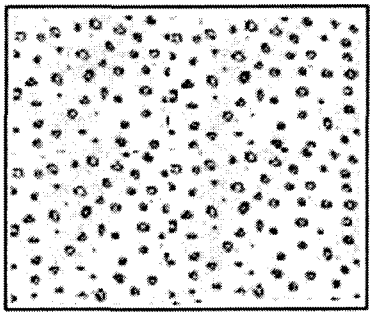

Figure 1-8, Schematic of a Mushy Fluid

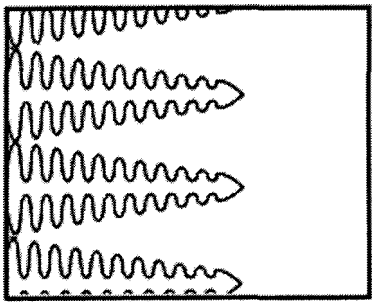

Figure 1-9, Schematic of a Columnar Zone

The only difference in above cases is in the velocity set for the solid or liquid. For example, for a mushy fluid model, $\vec{u} = \vec{u}_l = \vec{u}_s$ and for a columnar zone, in a static casting, where $\vec{u}_s = 0$, the velocity for the liquid will look like $\vec{u} = f_l \vec{u}_l$ [11]. One can base the type of process chose on these models.

1.2.3.2 - Shrinkage

Another difficulty in the modeling of a solidification problem is one associated with shrinkage. Usually material densities are different in their solid and liquids states. Changes in densities cause a change in volume which is referred to shrinkage. Shrinkage may be positive if the solid has a higher density and negative if the solid has lower density than the liquid state. When shrinkage is positive, during the solidification process, the difference between densities necessitates the melt to move toward the solidifying front. In the case of having negative shrinkage, the opposite scenario happens. Either way, flow is induced in the melt. This flow may have strong impact on the solidified composition and contribute to macrosegregation in the cast alloy, even though the

magnitude of the velocities produced by shrinkage is small, particularly in the dendrite and mushy zone [12].

Shrinkage can sometimes cause cracking to occur in the component as it solidifies. Since the coolest area of a volume of liquid is where it contacts a mold or die, solidification usually begins first at this surface [12]. As the crystals grow inward, the material continues to shrink. If the solid surface is too rigid and will not deform to accommodate the internal shrinkage, the stresses can become high enough to exceed the tensile strength of the material and cause a crack to form [12].

1.3- Summary

In this chapter, a general phase change phenomenon and related modeling issues including mushy region and shrinkage were explained. This brief summary of the critical features of modeling provides the physical background for this study and is important for the survey of modeling methods used.

1.4- Outline of thesis

The remainder of the thesis is structured as follows:

- Chapter 2: a background and literature review related to the numerical techniques and methods used in the current work is presented. At the end of chapter the scope of current work is introduced.
- Chapter 3: in this chapter, the concept of conservation laws which are used to construct the governing equation is introduced. Then, a finite volume discretisation scheme is used for the discretization of the governing equations. The discretized form of governing equation is presented.

- Chapter 4: VOF method is explained in details. The VOF reconstruction algorithm is summarized in steps. Finally, the solution procedure for a solidification problem is presented in this chapter.
- Chapter 5: this chapter includes all validation cases. To validate, the developed code is applied to three groups of problems. The first, only a coupled mass-momentum set for a virtual stationary liquid-solid interface defined by different values of f in the domain using VOF method is solved. The second, consisting of solving a linked energy equation-VOF reconstruction algorithm for a simple heat transfer without phase change. The last group includes a set of solidification problems.
- Chapter 6: The thesis is summarized, and the main conclusions and recommendations for future research are given.

Chapter 2

2. Background and Literature Review

In the following section the main intention is to present a literature review on three important aspects of numerical modeling of any phase change problem. At the beginning, a brief overview of fixed grids versus transformed grids approaches is presented, followed by differences between two popular interface treatment methods. At the end, a summary of the literature review is presented on the method which is chosen to treat the solidification interface in the present work.

2.1- Fixed grids versus Transformed grids

To solve two-phase problems with immiscible fluids, the incompressible Navier-Stokes equations need to be solved. The presence of moving sharp interfaces or fronts where pressure and velocity derivatives may have jump discontinuities cause the most difficulties in such problems. As the result, the central problem in the numerical modeling of phase change processes is the treatment of the heat and mass flow conditions that happen at the moving solid-liquid interface. A number of methods have been developed in recent years for the solution of problems involving moving interfaces in multiphase systems. These methods can be divided into two main classes depending on the type of grids used [5]: Transformed grids approach and fixed grids approach.

2.1.1- Transformed grids approach

In this approach, the governing equations and their boundary conditions are considered in a generalized curvilinear coordinate system. The equations can then be solved on a rectangular and uniform space grid, which remains fixed in space and time.

For example, in a simple 1D isothermal solidification problem, one can use the so-called Landau transformation, that is [13, 14],

$$\zeta = \frac{x}{S(t)} \tag{1-1}$$

to fix the physical front $S(t)$ on the line ζ for all time t. In more complex systems such as 2D problems, for instance, one can transfer the original coordinate system (x, y) into a new system (ζ, η) which is called body-fitted coordinate (BFC) [13,14]. Then this fixed (ζ, η) mesh, corresponding to the physical moving (x, y) mesh, can be used throughout the calculation. Basically, the movement of the interface and of the mesh points in the original region appears only as changes in the x and y at the corresponding, fixed (ζ, η) points at each time step [13, 14]. By using this approach the moving interface between solid and liquid is essentially immobilized [15] and is treated as a boundary between elementary domains. This approach allows a precise representation of the interfacial jump conditions, at least in principle [16, 17].

2.1.2- Fixed grids approach

In this approach a fixed grid is applied directly in the real problem domain. This approach uses fixed grids to describe the velocity field but requires specific advection schemes and source terms in order to preserve the sharpness of the interfacial front [15]. Lacroix and Voller, [15], compared the results of using these two approaches in a solidification problem. The authors mentioned that the major advantage of a transformed grid over a fixed grid approach is that; the governing flow and heat equations are solved on a simple fixed rectangular and uniformly spaced computational grid with no loss of accuracy in discretization near curved boundaries. However, the simplicity in the

numerical grids comes at the expense of much more complex governing equations in transformed grids.

The results of transformed and fixed grids approach were also compared to some experimental results by [15]. Both approaches give predictions in very close agreement [15]. With regard to CPU usage, the fixed grid method requires less CPU time at smaller time steps, however, at large time steps; the CPU minutes per time step for both approaches are similar [15].

One of the major drawbacks of fixed grid is, in some cases, accurate fixed-grid predictions can only be obtained with fine grids that are much denser than the mesh used in transformed grids.

Using fixed or transformed grids in a case like an isothermal phase change problem can depend on the availability of codes, expertise and personal preference. However, in cases involving complex problems that require resolution, one may consider the advantage and drawback of each approach and then decide. It was suggested by [15], and successfully applied by [18, 19], for problems that involve multiple fronts, i.e. solidification of binary alloys, the complexity of the problem is such that individuals prefer fixed grids over transformed grids approach. Furthermore, moving boundaries in the solidification case, may undergo large deformations and for this reason fixed grid techniques are advantageous and used in current work.

2.2-Treatment of the interface using fixed grids approach

As it was mentioned, choosing a fixed grid system requires preserving the sharpness of interface using special advection schemes. These advection schemes may in turn be divided into two groups, either implicitly or explicitly representing the interface.

Excellent overviews of the interface treatment methods are given by Hyman [20], Unverdi and Trygvason [21], Sussman and Smereka [22], Rider and Kothe [23], and Rudman [24, 25]. There are two popular methods used to approximate interfaces [26]. Detailed descriptions of the two methods follow next.

2.2.1- Interface Tracking Methods

Methods in which the interface is treated as a sharp interface, whose motion is followed, are labeled front (interface) "tracking" methods [26]. These methods explicitly track the interface either by marking it with special marker points, (See Glimm et al. [27], Daly [28], and Popinet and Zaleski [1], Hyman [20]), or by attaching it to a mesh surface which is forced to move with the interface. See Figure 2-1.

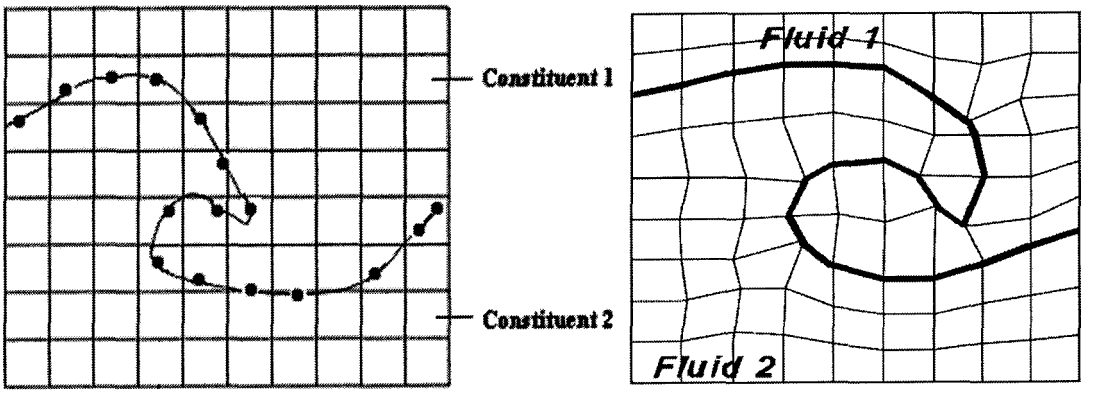

Figure 2-1, Schematic results of interface tracking method

For the case of using markers on the interface, the position of the interface is approximated by interpolation between these points, usually piecewise polynomial [20]. There are several ways of marking the interface, two of which will be explained here.

1) Particles on interface

Daly [28] presented a method to track an interface explicitly on a fixed grid by marking the interface with a group of connected mass-less marker particles. The local velocities are used to advect the mass-less particles in a lagrangian manner.

This method is sensitive to the spacing between the marker particles. If the markers are too close, then local fluctuation in the new positions of the particles can increase to a very high interface curvature resulting in strong surface tension forces. If they are far apart from each other, the interface is not well resolved. In this method, the interface particles do not retain their spacing throughout the calculation as the interface is moving and it is necessary to add or delete marker particles dynamically. Also, for the calculation of the interface curvature the order of the particles needs to be sequential along the interface [28, 29].

The idea of having sequential fixed particles is a major disadvantage of this method because it limits the prediction of merging or rupturing interfaces. Furthermore, in three dimensions, bookkeeping of the particles becomes almost impossible [29].

2) Height Function

Nichols & Hirt [30] extend the idea of setting markers on the interface by relating the reference points on the interface to points on a certain plane. The location of the interface is then approximated by its height or distance from the reference plane [30]. Figure 2-2 gives a schematic of the height function implementation for open interfaces.

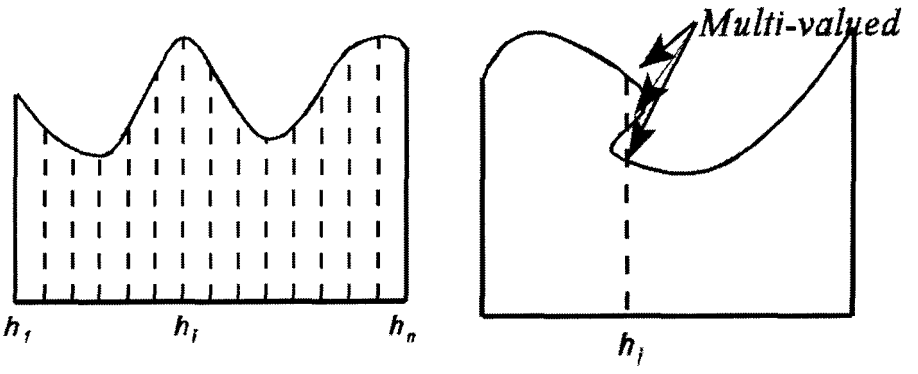

Figure 2-2, Schematic of height function for open interfaces

A major limitation of this approach is that every reference coordinate can represent only one interface value. Therefore, it is not possible to predict the situation where the reference coordinate can be multi valued See Figure 2-2. However, this method is efficient in terms of computer storage and suitable for non-complex interfaces in a three dimensional space [30].

Having talked about marker points, also, various techniques have been developed to attach the interface to a mesh surface during past decades including *Interface (surface) fitted method.* (Ferziger & Peric [26], Dervieux & Thomasset [31], Glimm *et al.* [32, 33])

These methods are mainly created to reduce computer storage needed for the interface markers method and to ensure the existence of a sharp interface [26]. These methods are limited to problems where the interface is not subjected to large deformations, because these lead to significant distortion of the mesh. If one uses these methods in a problem with a high level of deformation happening at the interface, it is necessary to re-mesh continuously, which brings more complexities to the problem [26].

2.2.2- Interface Capturing Methods

The second category of interface approximation is so called *"front-capturing-methods"* in which the location of the interface is not defined as a sharp boundary [26]. The computation is usually performed on the fixed grid and the shape of the interface is determined by computing the fraction of each near-interface cell that is partially filled.

For example, this can be achieved by (1) a level-set-method which relies on an implicit description of the interface, given through a distance function [34], or introducing a set of mass-less particles at the interface and following their motion, (2) marker and cell, (MAC), a method proposed by Harlow and Welch [35]. Alternatively this can be achieved by solving a transport equation for the (3) fraction of the cell filled by the liquid-gas or any other constituent (VOF), a method that proposed by Hirt and Nichols [36]; See figure 2-3. Below is more detail about each of these methods:

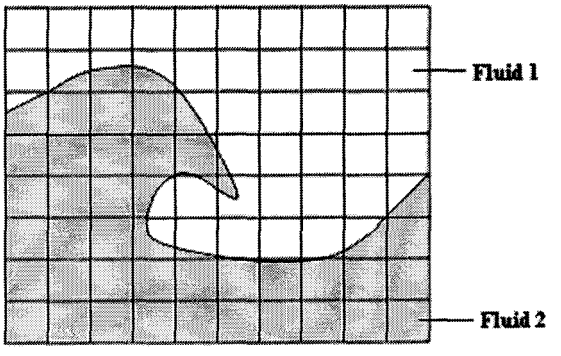

Figure 2-3. Schematic result of Interface capturing schemes

1) Level-set method

In this approach, a continuous function, known as a level-set function is introduced over the whole computational domain. At each point, the value of level-set function is defined as the shortest distance between that point and the interface [22]. Where this

function has the value of zero, the interface is defined. To make a distinction between the two constituents on either side of interface a negative sign is set to the distance function for one of the constituents [22]. The level-set method has been initially used for the modelling of flame advection [37]. Sussman et al. [38] used this approach to model the motion of bubbles and droplets. Although this approach is applied in different problems, the significant drawback of the method is the existence of a steep gradient in the level set function when two interfaces are merging together [38]. The difficulties caused by using this method have been removed partially in past years, particularly by Sussman et al. [38].

These methods are somewhat complex to implement, but give the precise location and geometry of the interface [26].

2) Particles in fluid or Marker and Cell (MAC)

In the MAC method due to Harlow and Welch (1965), [35], as it was mentioned, a set of mass-less marker particles are spread over the volume occupied by one of the constituents, usually the one with the free surface. A cell with no markers is considered to be empty. A cell adjacent to empty cells contains a segment of the interface. All other cells with marker particles are considered to be filled by another constituent.

The marker particles are used to distinguish between the two constituents and do not participate directly in the calculation. These particles move according to the velocity components in their vicinities and serve as a flow identification aid. Later, Daly [28] extended the MAC method of Harlow and Welch [35] to deal with a two fluid problem.

In Daly's method, the marker particles are not only used to distinguish between the two fluids or constituents but also to determine the density and viscosity of the mixture in each cell. Figure 2-4 shows a schematic of the MAC method.

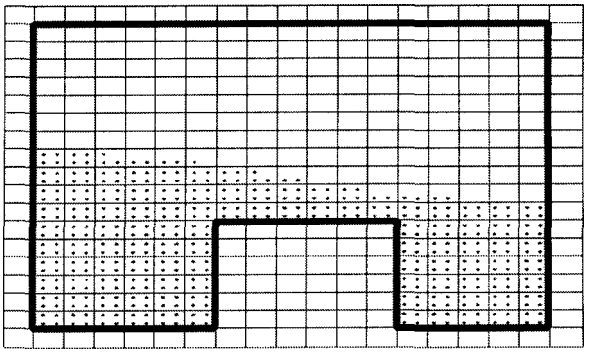

Figure 2-4, Schematic of MAC method

The MAC method is readily expandable to three-dimensional computations and also can handle complex phenomena like wave breaking. However, in three-dimensional calculations, since the motion of a large number of particles needs to be followed in addition to solving the equations governing the flow, a significant increase in computational effort is required [26].

3) Volume-fraction

In this method, to distinguish between two different constituents in the domain, a scalar indicator function between zero and one, known as the volume fraction f is used.

The value of unity indicates one constituent and zero another. A value between zero and unity in a computational cell indicates the presence of an interface in that cell and the value itself indicates the relative proportions of the constituents occupying the cell volume. If one uses, for example, volume fraction of fluid as the base of calculations, the method is called *Volume-of- Fluid* method. Since in volume-fraction methods only one

value needs to be assigned to each cell, this method is much more economical than MAC. The Volume-fraction method suffers from smearing of the interface over meshes due to using common convective differencing schemes.

Many multiphase flow problems of interest involve constituents with very high ratios of density and viscosity. In such problems, surface tension effects or/and pressure jump effects are usually large compared with viscous damping. In these cases, the effect of the unbalanced forces acting on the interface not only reduces the accuracy but also can lead to "spurious currents" [1]. Figure 2-5 shows the spurious current reported by Popinet and Zaleski [1] due to large surface tension effects. Currents similar to this are created at solid-liquid interface.

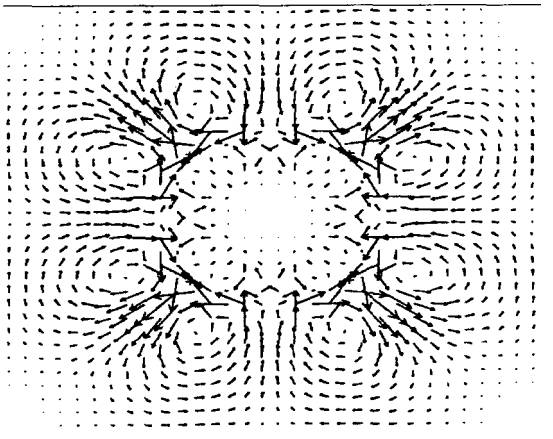

Figure 2-5, Spurious current around a stationary bubble. The method used for the interface advection is a VOF method explained in [1]

In general, implementation of front capturing methods on fixed grids is less complex than front-tracking methods, however, one has to remove the numerical artifacts like "spurious currents", that may result from inconsistent modeling of surface tension, in case of gas-liquid or liquid-liquid phases in the domain, and the associated pressure jump [1].

In this work a Volume-of-Fluid based method is used to capture the interface between the liquid and solid phases. The next section is basically an overview of techniques that have been developed to maintain a well defined interface within the volume-fraction framework and particularly for the Volume-of Fluid method.

2.3- An overview on Volume-of-Fluid (VOF) methods

Volume-of-Fluid methods have been in use for several years. Volume-of-fluid methods automatically handle changes in the global topology of the front, such as fronts that break up into droplets or fronts that collide with themselves and merge. This eliminates the algorithmic complexity that can occur when the front is modeled by a collection of line segments or polygons. Furthermore, the logical structure of the algorithm is not significantly more complicated in three dimensions than in two. The most well-known volume-of-fluid algorithm is the VOF algorithm of Hirt and Nichols [36]. Several codes based on this VOF algorithm, namely SOLA-VOF [36, 39], NASA-VOF2D [40], RIPPLE [41, 42] have been, and continue to be, widely used by researchers to model interfaces in industrial applications. The Volume-of-Fluid methods in all of these cases are built on a relatively crude interface reconstruction algorithm that relies on a piecewise constant representation [43] of the interface, or piece-wise linear approximation using the constituent volume fraction evolution equation [44, 23].

Generally, a Volume-of-Fluid algorithm evolves the constituent volumes by initializing and updating a volume fraction field that identifies the constituent contained in each computational cell, and the interface is reconstructed from this volume-fraction field, as required. The reconstructed interface is not unique, as it depends on the reconstruction techniques.

DeBar [45] was one of the earliest persons who implemented a Volume-of-Fluid algorithm in a two-dimensional Eulerian method with a low accuracy, essentially first–order accurate, piecewise constant reconstruction technique, to model compressible multi-phase flow. A simple Line Interface Calculation (SLIC) technique was one of the early methods for defining geometrical approximation to the fluid interface, which was used by Noh and Woodward [43]. The SLIC method has been used very widely in shock wave refraction at a gas interface [46, 47]. The original SLIC method of Noh and Woodward approximates the interface in each cell as a line parallel to one of the coordinate axes and assumes different fluid configuration in that cell for the horizontal and vertical movement respectively. Their version of SLIC is a strictly one-dimensional method in which one uses the information in a 3×1 block of cells. In other words, for the x-sweep the volume fraction values on the left and right side of a cell are used to approximate the fluid or constituent distribution within the mesh cell. Values of above and below the cell are used for the y-sweep in a manner similar to the x-sweep. More details about SLIC can be found in [48]. Chorin [49] improved on SLIC method by using all the direct neighbors for the approximation of the fluid distribution in an interface (mixed) cell and by introducing a fifth type of building block, namely a corner element.

This approach, similar to the original SLIC method, also takes the direction of motion into account and gives different fluid distribution in a cell for different sweeps directions [49].

Youngs [50] introduced a useful refinement to the SLIC method with the use of oblique lines to approximate the interface in a cell, also known as Piecewise-Linear Interface Calculation (PLIC), (Figure 2-6). An improvement of Poo and Ashgriz [51], the

Flux-line-segment Model for Advection and Interface reconstruction (FLAIR), is also considerable.

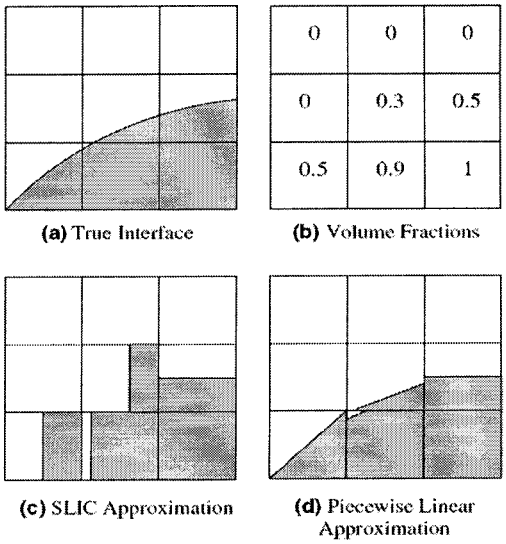

Figure 2-6, Schematic of different reconstruction schemes

Later, Pilliod and Puckett [48] demonstrated that a sufficient condition for a Volume-of-Fluid interface reconstruction algorithm to be second-order accurate on smooth interfaces is for the algorithm to reproduce linear interfaces exactly.

They proposed two interface reconstruction algorithms; the Least Squares Volume-of-Fluid Interface Reconstruction Algorithm (LVIRA) and the Efficient Least Squares Volume-of-Fluid Interface Reconstruction Algorithm (ELVIRA) both have this property.

These second-order accurate piecewise-linear interface reconstruction algorithms have been used widely to model a variety of compressible and incompressible flows [52, 36]. In the present work the LVIRA method is used to reconstruct the interface. The details of this method will be explained later in section 4-4.

As was previously mentioned, one of the crucial elements in VOF methods is how to accurately update the volume-fraction field so as to match the evolution of the fluids or constituents. In general, in VOF methods the interfaces are tracked by evolving fluid volumes forward in time with solutions of an advection equation [23, 34, 50-53]. See Appendix A for details.

In a general Volume-of-Fluid method, as was mentioned before, interface geometry must result from the local volume data and the assumption of a particular algorithm, before the interface can be reconstructed. Then, the reconstructed interface is used to compute the volume fluxes necessary to integrate the volume evolution equations.

All works in which a VOF method has been used, so far, were related to cases in which either two immiscible fluids were considered[21,23-25,29,34,38,43,47], or a fluid and gas comprised the full domain [1,16,20,22,23,27,28,30,36,46], and the objective of the problem was to capture the interface between the two fluids or capture a free surface to be able to manage interface changes. In such problems, the flow field was strong and rarely involved a phase change problem. As a result, using an advection equation for the evolving volume fraction of the constituent using the fluid velocity field was reasonable.

However, in the current work, the solidification process involves solid-liquid as the two constituents in the domain, and much of the solidification process takes place after the high-speed shot into the mold has occurred. Thus, the fluid motion is due mainly to convection currents and to the shrinkage that occurs during solidification. The evolving volume fraction of melted alloy based on this weak flow velocity in the cavity may not produce a terribly accurate update of the volume-fraction of the fluid. Instead,

here a different approach is presented. The author has attempted to create a robust, accurate and easy platform to be upgraded to three-dimensional or axisymmetric cases.

2.4- Summary

At this point, the author has chosen a fixed grid approach and a VOF method for interface capturing. To meet these requirements, the present algorithm comprises the following:

- A VOF in-house code is developed and implemented for the LVIRA reconstruction algorithm for a solid-liquid phase change problem
- A new pressure approximation is presented to overcome the existence of "spurious currents" near the interface due to pressure jumps.
- A link between the source based energy equation of Voller et al. [54] and VOF reconstruction method is described and implemented.

Chapter 3

3. Mathematical Model

Due to the absorption or release of latent heat and the presence of a complex interfacial structure that characterizes the phase change of most materials, exact solutions describing this process are not yet possible [55].

Most of the numerical works used to date use either a *multiple region* or a *single region* formulation, depending on the type of grid used; *Transformed or Fixed.* In a majority of the works in which a transformed grid is implemented, the focus is on applying the multiple region formulation in which independent conservation equations are utilized for each phase. In this method, the mesh is continuously updated so that it always coincides with the phase change interface. These methods are generally easy to apply to pure materials involving one or two phases in one space dimension. It is a very difficult task, however, to apply the multiple region formulation to a case with fluid flow involved in a two dimensional problem. This is even more difficult in three dimensional cases [6].

The alternative is to come up with a form of equation, which applies over the entire region of a fixed domain. On the basis of the approach chosen herein, this is the method used. These methods are called *single region formulations* in which a system of conservation equations is utilized to be applied equally to both phases [6]. In these methods, no explicit conditions on the interface are required and the numerical solution can be carried out on a fixed grid [6]. Single region formulations are well suited for

modeling the phase change of mixtures where the latent heat is evolved over a range of temperature. The major advantage of single region formulations is that their solution can be obtained by conventional numerical methods. One must notice that, however, in order to predict the complex interfacial structures, particular attention must be directed to the handling of the latent heat evolution associated with phase change, as well as the numerical scheme utilized in the solution of the equations. Since a large number of articles have been published in this area, and also this kind of formulation is used in the current work, in this section, three prevalent methods are reviewed and then a comparison that leads to the final formulation is presented.

- **Apparent heat capacity method**

In this technique the latent heat is included in the specific heat of the material. An apparent heat capacity is defined in the range of temperature that phase change happens, so it accounts for the entire enthalpy change, including sensible and latent heat.

For the case of a pure material or an alloy of eutectic composition, an artificial phase change temperature range must be used. In this range, the latent heat is assumed to be released. One can review [56, 57] for more information.

According to Saludean, and Abdullah, [58], an important drawback of this method is that if, for a solidification problem, a nodal temperature in one time step falls from above the liquidus temperature to below solidus temperature, then for this particular node the latent heat released is not accounted for. Poirrier and Salcudan [59] showed that this issue happens in melting processes too. Saludean and Abdullah [58] show that, in order to solve this issue, a large mushy region has to be assumed, which results in destroying the physics of the problem.

- **Enthalpy method**

The enthalpy method has a lot more popularity than the apparent heat capacity method. Many authors used this method in different applications [60-64]. The basic method includes writing the transient part of the energy equation in terms of enthalpy instead of temperature. Then, based on the latent heat release characteristics of the phase change material, a variation between the total enthalpy and temperature is defined. This variation in the case of isothermal solidification is a step function and in case of binary alloy phase change, is usually assumed to be a piecewise linear function within the mushy region and a step function at the solidus temperature. The best detailed information can be found in [65]. The enthalpy method is reasonably accurate for materials solidifying over a range of temperature [59]. The solution is independent of the range of temperature and the time steps; however, it is more complex and more expensive than other methods along with the fact that the computational cost rapidly increases with mesh refinement. This model also performs poorly close to the interface for cases of isothermal solidification [62-64].

- **Source-based method**

A substitute to setting up a non-linear coefficient in the form of a specific heat, similar to the apparent heat capacity method, is to develop a non-linear source term in the energy transport equation. This technique is called a 'Source-based' technique [54]. This method can deal with a general liquid fraction curve, unlike the apparent heat capacity method which requires introducing a temperature range to ensure that the relation between the constituent volume fraction and temperature is at least piecewise continuous throughout the temperature domain. This technique also has certain similarity to the

enthalpy formulation. The latent heat appears as a source term in the energy equation and in this way the latent heat is directly coupled to the nodal temperature, which produces fairly accurate results, particularly for non-isothermal solidification. In this technique, the source term includes the unknown constituent volume fraction along with enthalpy or temperature in the governing equation. This problem can be removed by performing appropriate iteration [54].

- **Source based method versus Apparent heat capacity method**

A general understanding of the differences between source based methods and apparent heat capacity methods can lead one to the best choice of methods. To compare these two methods, it is best to consider a matrix form of the formulation used for the Source-based method and apparent heat capacity method. Considering a general discretized energy equation:

$$a_P T_P = \sum_{nb} a_{nb} T_{nb} + b_P T_P^{old} + V_p \delta H_P (f_{l_p}^{old} - f_l) \tag{3-1a}$$

For the case of a source-based method, the general matrix form can be written as:

$$(C + \Delta t K)T = CT^{Old} + M(f_l^{Old} - f_l) \tag{3-1b}$$

where M is a diagonal matrix with elements

$$M_{PP} = V_P \delta H_P \qquad (3\text{-}1c)$$
$$M_{ij} = 0 \text{ if } i \neq j$$

and the relationship between matrix and final discretized form scheme are

$$\begin{cases} K_{PP} = a_P \\ K_{Pj} = a_{nb} \\ j = 1,2,...,P-1,P+1,...,n-1,n \\ C_{pp} = b_P \end{cases} \qquad (3\text{-}1d)$$

and in case of apparent heat capacity :

$$(C + \Delta t K)T = CT^{Old} \qquad (3\text{-}2)$$

In the apparent heat capacity method, the source term is dropped and the diagonal element of C is given by

$$C_{ii} = C^{apparent} V_P = \left(\rho C_P + \frac{dF}{dT} \delta H_P \right) \times V_P \qquad (3\text{-}3)$$

If $F(T)$, a liquid-fraction temperature relationship, is steep then the apparent heat capacity can be large and this will stabilize the iterative solution of the discrete equation due to slower change in temperature in cells where phase change is occurring. The stability can be considered an advantage of the apparent heat capacity method over the source-based method. However, when $F(T)$ is relatively steeper or contains a jump, the apparent heat capacity method fails. Corrective schemes and modification are necessary to overcome such problems. Such modification and remedial schemes are not required in

Source-based methods, which, though, suffer from convergence problems. To overcome these convergence problems, a new source-based method was introduced by Voller and Swaminathan [66]. The motivation of the new source based method of Voller was to retain all of advantageous of the apparent heat capacity and previous source-based methods without incurring any of the disadvantageous. The energy equation used in the present work is similar to the source-based energy equation proposed by Voller and Swaminathan [66].The derivation of the source-based energy equation is outlined in the following sections.

3.1- The Governing Transport Equations

Problems under consideration are those of fluid flow and heat transfer in a domain consisting of fluid, mushy, and solid regions. The behavior of a phase change system can be described by the conservation equations for mass, momentum and energy in the solid, liquid and mushy zones. In addition, appropriate relations are required for determining the portion of solid or liquid in each computational cell as a function of temperature.

Also, this is necessary for representing the variation of mixture properties in the mushy region.

In the following sections, under the assumption of Newtonian laminar flow, constant densities except for the buoyancy terms, negligible shrinkage-caused flow, local thermodynamic equilibrium and negligible species diffusion in the solid, the details of the flow and heat transfer governing equations will be introduced.

3.1.1-Mass and Momentum Equations

In Cartesian notation the continuity and conservation of momentum equations may be written as

Continuity:

$$\left(\nabla \cdot \rho \vec{u}\right) = 0 \tag{3-4}$$

x-y-momentum:

$$\frac{\partial}{\partial t}(\rho u) + \nabla.(\rho \vec{u} u) = -\frac{\partial P}{\partial x} + \nabla.(\mu \nabla u) + S_u \tag{3-5a}$$

$$\frac{\partial}{\partial t}(\rho v) + \nabla.(\rho \vec{u} v) = -\frac{\partial P}{\partial x} + \nabla.(\mu \nabla v) + S_b + S_v \tag{3-5b}$$

Or in more general form of:

$$\frac{\partial \rho \vec{u}}{\partial t} + \nabla.(\rho \vec{u}\vec{u}) = -\nabla.P + \nabla.(\mu \nabla \vec{u}) + S \tag{3-6}$$

where P is pressure, ρ is the density, μ is the dynamic viscosity, $\vec{u} = (u, v)$ and S_u, S_v are source terms which will be defined later. S_b is the volumetric source term, assuming the Boussinesq treatment to be valid, i.e., density is constant in all terms except in the gravity source term, then natural-convection effects can be accounted for by defining the buoyancy source term to be

$$S_b = \rho g \beta (T - T_{ref}) \tag{3-7}$$

where β is the volumetric thermal expansion of the material.

Volume tracking, particularly, interface capturing methods, use a characteristic function, F, associated with the portion of the cell that is occupied by one of the constituents, which is fluid in the current work ; i.e., in a cell with the dimension of, $\Delta x \times \Delta y$,

$$f_{i,j}\Delta x\Delta y = \text{Volume of fluid in cell } (i,j)$$

The number $f_{i,j}$ is called the volume fraction (of fluid) in the (i,j)th cell, where $0 \le f_{i,j} \le 1$. The volume fraction of the other material, which is solid in the current work, is $(1-f_{i,j})$. The discretization of the characteristic function associated with the fluid considers the fact that,

$$f_{i,j}\Delta x\Delta y \approx \iint_{i,jcell} f(x,y)dxdy \text{ , is } f(x,y) = \begin{cases} 1 & \text{if there is fluid at point (x, y)} \\ 0 & \text{if there is solid at point(x, y)} \end{cases}.$$

Any value of f between zero and one indicates a mixed cell and a mushy region in the domain.

The density and dynamic viscosity in the momentum equations are related to the individual constituent's properties via the characteristic functions:

$$\rho = f\rho_l + (1-f)\rho_s \tag{3-8}$$

$$\mu = f\mu_l + (1-f)\mu_s \tag{3-9}$$

where the subscripts l and s refer to liquid and solid phase, respectively and μ_s can be set to a large number.

The condition that all velocities in solid regions are zero must be accounted for, somehow, in the momentum equation. The basic principle is to gradually reduce the velocities from a finite value in the liquid zone to zero in the pure solid zone, over the computational cells that are changing phase. To define a source term in momentum equations, one has to decide about the model for the mushy region formed in the domain.

In the current work, since a "columnar zone" model works best for most of alloys, it is assumed that the mushy region follows a "columnar zone" shape. Then, it can be assumed that the mushy region behaves as a porous media [67]. As a result, the source term must be defined so that the momentum equations are forced to mimic the Carman-Kozeny equation for flow in a porous media [10]:

$$\nabla P = \left[-\kappa \frac{(1-\varepsilon)^2}{\varepsilon^3} \right] \vec{u} \tag{3-10}$$

Here, $\varepsilon, \vec{u}$ are porosity and the field velocity, respectivly. κ is related to the medium permeability and is accounting for the mushy region morphology in the current work. In order to achieve such behavior, an appropriate definition of the source term (for example for u-momentum) will be:

$$S_u = \left[-\kappa \frac{(1-f_l)^2}{f_l^3 + A} \right] u \tag{3-11}$$

where A is merely a computational constant introduced to avoid division by zero. As it can be understood, the effect of source term would be as follows:

In full liquid cells, where $f_l = 1.0$, S_u is zero and has no influence; in cells that are changing phase, the value of S_u dominates over the transient, convective and diffusive components of the momentum equation, thereby the Carman-Kozeny law is satisfied; in total solid cells, where $f_l = 0.0$, the resulting large value of S_u will swamp out all terms in the momentum equations and force any velocity predictions effectively to zero. It is worth mentioning that the validity of this model was experimentally

investigated and showed comparable results to numerical results [65]. A similar source term can be defined in y-momentum equation.

3.1.2- Energy Equation

Ni and Beckermann[68] derived an appropriate form of the governing energy equation using a local volume averaged model of the two phase solid and liquid region. To derive the equation, a small 'representative elementary volume', REV, was chosen and microscopic balances were performed on the enthalpy transport in distinct liquid and solid parts within the REV. Assuming a constant REV temperature $T = T_s = T_l$, thermal equilibrium, and small local velocity variations, averaged equations were obtained,[3,55]

$$\frac{\partial}{\partial t}(f_s H_s) + \nabla.(f_s H_s \vec{u}_s) = \nabla.(f_s k_s \nabla T) - (\text{interface terms})_s \tag{3-12}$$

$$\frac{\partial}{\partial t}(f_l H_l) + \nabla.(f_l H_l \vec{u}_l) = \nabla.(f_l k_l \nabla T) + (\text{interface terms})_l \tag{3-13}$$

Since the REV is assumed to be isothermal, specific enthalpies of solid and liquid, H_s and H_l, can be defined as:

$$H_s = \int_{T_{ref}}^{T} \rho_s C_s d\theta \tag{3-14}$$

$$H_l = \int_{T_{ref}}^{T} \rho_l C_l d\theta + \rho_l L_{fs} \tag{3-15}$$

where, C_s and C_f are the specific heats of solid and liquid, respectively, and L_{fs} is the latent heat of phase change. On assuming that only two phases are present, meaning that void formation and such similar phenomenon are neglected, $f_s = 1 - f_l$.

It is convenient to combine equations (3-12) and (3-13) by adding them together [54, 66] to attain a single energy equation. By adding these two equations, interface terms will cancel out and a single governing equation results:

$$\frac{\partial H}{\partial t} + \nabla.(f_s H_s \vec{u}_s + f_l H_l \vec{u}_l) = \nabla.(k_{mix} \nabla T) \tag{3-16}$$

On the other hand, integrating (3-14) and (3-15) will result in

$$H = \rho C_s (1 - f_l)(T - T_{ref}) + \rho f_l C_l (T - T_{ref}) + \rho f_l L_{fs} \tag{3-17}$$

and on using equation (3-17),

$$\frac{\partial H}{\partial t} = \{\rho(1 - f_l)C_s + \rho f_l C_l\}\frac{\partial T}{\partial t} + \{\rho(C_l - C_s)(T - T_{ref}) + \rho L_{fs}\}\frac{\partial f_l}{\partial t} \tag{3-18}$$

Defining C as a mixture specific heat capacity,

$$C_{mix} = C_s(1 - f_l) + C_l f_l \tag{3-19}$$

and

$$\delta H = \{\rho(C_l - C_s)(T - T_{ref}) + \rho L_{fs}\} \tag{3-20}$$

the form of the energy equation, assuming, $T_{ref} = 0$, will be derived as:

$$\rho C_{mix} \frac{\partial T}{\partial t} + \nabla.(f_s H_s \vec{u}_s + f_l H_l \vec{u}_l) = \nabla.(k_{mix} \nabla T) - \delta H \frac{\partial f_l}{\partial t} \tag{3-21}$$

where,

$$k_{mix} = (1 - f_l)k_s + f_l k_l \tag{3-22}$$

Considering the fact that velocity in the solid region is zero and substituting equation (3-15), the final form of the energy equation will be:

$$\rho C_{mix} \frac{\partial T}{\partial t} + \nabla.(\rho C_l \vec{u} T) = \nabla.(k_{mix} \nabla T) + S_T \tag{3-23}$$

where, S_T, is a source term defined as:

$$S_T = -\delta H \frac{\partial f_l}{\partial t} - \nabla.(\rho L_{fs} \vec{u}) \tag{3-24}$$

One of the key features in the solution of this kind of energy equation (3-23) is the definition of the local volume fraction. In general, the local liquid volume fraction depends on the nature of solidification [66]. The local volume fraction could be a function of temperature, cooling rate, speed of solidification and the rate of nucleation if the kinetics of liquid-solid transformation is such that under-cooling is considerable [69,70]. In a system of multi-component alloys, solutal transport (macro-segregation) will also influence the local liquid volume fraction field [11]. For simplification, however, it is convenient to assume the liquid volume fraction is only a function of temperature [66].

Authors of [66] demonstrated different possible forms for liquid fraction variation versus temperature functions including: linear function, linear eutectic function, power function and Scheil equation in their work. In the current work, the fraction of liquid in the mushy region is estimated by the Scheil equation, which assumes perfect mixing in the liquid and no solid diffusion. With the liquidus and solidus having constant slopes, f_l can be expressed as [66, 71]:

$$F(T) = f_l = \left(\frac{T_m - T}{T_m - T_l}\right)^{\frac{1}{k_p - 1}} \tag{3-25}$$

where T_m is the melting point, T_l is the liquidus temperature and k_p is the equilibrium partition ratio.

A schematic of the Scheil equation, which illustrates the liquid fraction variation versus temperature, is shown in Figure 3-1.

Regarding this graph, one may notice

1. a temperature range associated with a non-linear change of f_l in the mushy region.
2. a step discontinuity at the mush/ solid interface that can be associated with an eutectic phase change [54]; and also
3. a step discontinuity at the mushy/liquid, which can be associated with under-cooling at the dendrite tips [54].

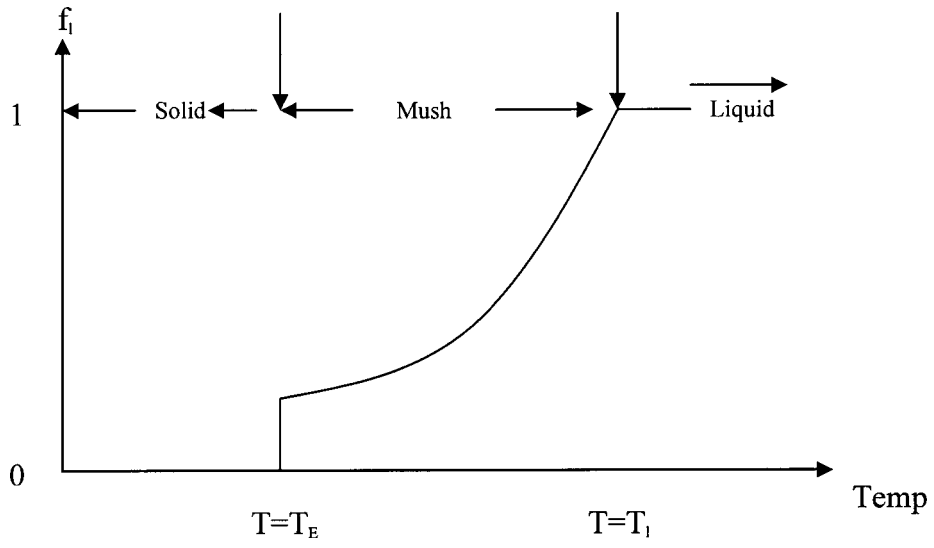

Figure 3-1, Schematic of liquid volume fraction VS. Temperature

3.2- Discrete Governing Equations

A summary of the proposed discretization of the governing equations is presented in this section. The discretization is carried out with using a co-located finite volume method (FVM) [26, 72] implemented on an orthogonal rectilinear grid. A portion of this grid is shown in Figure3-2.

The calculation domain is first sub-divided into rectangular control volumes, or cells, and the grid points (nodes) are placed at the centers of the cells. All dependent variables are stored at the nodes of the computational mesh.

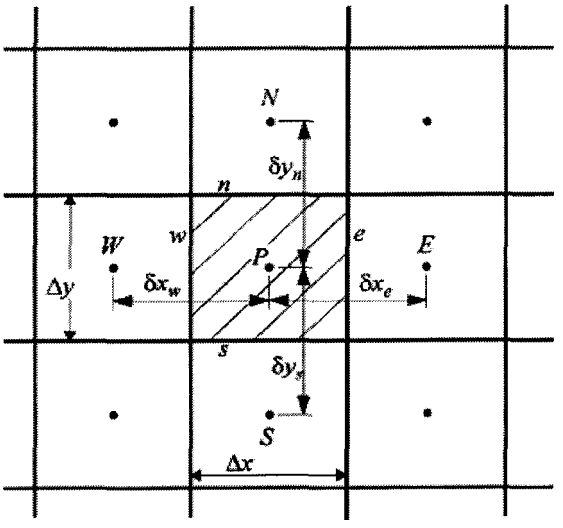

Figure 3-2, A portion of the grid and related nomenclature used in the FVM

3.2.1- Mass and Momentum equations

In order to attain the discretized form of the governing equations presented in the previous section for solution in a collocated finite volume framework, these equations must be integrated over a typical control volume, and time. Considering a volume V bounded by N faces and centered about a node P, the partially discretized forms of

Equations (3-4) and (3-6) are given, respectively, for a fully implicit discretization as [72]:

$$\sum_i \dot{m}_i = 0 \tag{3-26a}$$

$$\frac{\rho V_P\left(\vec{u}_P - \vec{u}_P^{Old}\right)}{\Delta t} + \sum_i \dot{m}_i\left(\vec{u}_i - \vec{u}_P\right) = -V_P(\nabla P)_P + \sum_i \mu_{mix}\left(A\frac{\partial \vec{u}}{\partial n}\right)_i + S_b V_p + S V_p \tag{3-27}$$

In Equations (3-26a) and (3-27), terms with the subscript i, i = 1, 2,..., N, are evaluated at the integration point located at the centroid of face i. Note that Equation (3-27) is in fact obtained by subtracting Equation (3-26a), multiplied by $\vec{u}_P$, from the discretized form of Equations (3-6). This is done to ensure a conservative method. The values of the advected properties $\vec{u}_i$ may be determined in the interior of the region from any convenient numerical scheme, as may the velocity gradients and the cell-centered pressure gradient. The mass flow rate is given by:

$$\dot{m}_i = \rho A_i\left(\vec{\hat{u}}_i.\hat{n}\right) \tag{3-26b}$$

where the advecting velocity $\left(\vec{\hat{u}}_i.\hat{n}\right)$ appearing in the mass flow rate expression for the CV faces is adopted in accordance with the co-located variable method of Rhie and Chow [73] to avoid check-board pressure distributions.

As an example, consider center node (P) of the simple grid illustrated in Figure 3-2, and consider the advecting velocity $\vec{\hat{u}}_e$ at the integration point e located on the east face of the volume associated with that node. It must be noticed that upon using

appropriate approximations for the face velocity gradients and advected velocities, the component of Equation (3-27) in the x-direction may be written as [72]:

$$a_P u_P = \sum_{nb} a_{nb} u_{nb} + b_P - V_P \left.\frac{\partial P}{\partial x}\right|_P = \tilde{u}_P - V_P \left.\frac{\partial P}{\partial x}\right|_P \tag{3-28}$$

Using the same analogy, a similar equation can be obtained at the E node:

$$a_E u_E = \tilde{u}_E - V_E \left.\frac{\partial P}{\partial x}\right|_E \tag{3-29}$$

Then, an estimate for $\hat{u}_e$ is attained by constructing an approximate momentum equation in the form of Equation (3-29) about the integration point e.

$$a_e \vec{\hat{u}}_e = \tilde{u}_e - V_e \left.\frac{\partial P}{\partial x}\right|_e \tag{3-30}$$

Here,

$$\vec{\hat{u}}_e \approx \frac{1}{2}(\tilde{u}_P + \tilde{u}_E) = \frac{1}{2}(a_P \vec{u}_P + a_E \vec{u}_E + \left.\frac{\partial P}{\partial x}\right|_P V_P + \left.\frac{\partial P}{\partial x}\right|_E V_E) \tag{3-31}$$

More approximations $a_e \approx a_P \approx a_E$ and $V_e \approx V_P \approx V_E$ are employed and note that here, a_E is the active coefficient for the node (E) in its own momentum equation. Finally, this leads to:

$$\vec{\hat{u}}_e = \frac{u_P + u_E}{2} - \frac{V_e}{a_e}\left(\left.\frac{\partial P}{\partial x}\right|_e - \frac{1}{2}(\left.\frac{\partial P}{\partial x}\right|_P + \left.\frac{\partial P}{\partial x}\right|_E)\right) \tag{3-32}$$

The first pressure gradient term in Equation (3-32) is a centered-difference approximation based on the values surrounding the *e* integration point and is treated implicitly in Equation (3-26b), while the second is the approximation based on the cell-centered values at *P* and *E*. In the y direction, all terms are similarly derived and cast in a form akin to that of Equations (3-29 to 3-32).

3.2.2- Energy equation

The generic discretization equation corresponding to the governing Equation (3-23), based on the fully-implicit time integration, subject to appropriate boundary and initial conditions, can be written as

$$\rho C_{mix}\frac{T_P - T_P^{Old}}{\Delta t}V_P + \sum_i \dot{m}_i(T_i - T_P) = \sum_i k_{mix}\left(A\frac{\partial T}{\partial n}\right)_i + S_T \qquad (3\text{-}33a)$$

In a more general form:

$$a_P T_P = \sum_{nb} a_{nb}T_{nb} + b_P T_P^{Old} + V_P \delta H_P (f_{lP}^{Old} - f_{lP}) + (convection\ Source\ term) \qquad (3\text{-}33b)$$

Note that to ensure a conservative method, Equation (3-33) is obtained the same way as momentum equations. Only differences appear in source term treatments.

There are two important factors that one may notice about the new source-based energy equation, (3-23):

1- The source term includes a transient term and a convective term.

2- The energy equation has two unknown variables, the liquid volume fraction, f_l, along with the temperature, T.

One of the principles behind the source-based method is to understand that the nature of the phase change process is characterized by the local liquid fraction field f_l. Basically, the nodal values of f_l keep track of the solid-liquid-mushy region as they change. An iterative solution strategy based on Equation (3-33) can be written as

$$a_P T_P^{m+1} = \sum_{nb} a_{nb} T_{nb} + b_P T_P^{Old} + S_T \tag{3-34}$$

where,

$$S_T = \underbrace{V_P \delta H_P (f_{lP}^{Old} - f_{lP}^{m+1})}_{Transient\ Source\ term\ S_{T_t}} + \underbrace{(convection\ Source\ term)}_{S_{T_C}} \times V_P \tag{3-35}$$

Term f_P^{m+1} causes non-linearity to equation (3-34). By using a truncated Taylor series expansion for this term;

$$f_{lP}^{m+1} = f_{lP}^{m} + \left.\frac{dF}{dT}\right|_{fl_P^m} [T_P^{m+1} - F^{-1}(f_{lP}^{m})] \tag{3-36}$$

where F is a function that characterizes the relation between T and f_l, and in the current work follows Equation (3-25). Using this linearization, only the transient source term looks like;

$$S_{T_t} = Q_P + R_P \times T_P^{m+1} =$$

$$\underbrace{\frac{-\delta H_P \times V_P}{\Delta t} \times \left\{ f_{lP}^{m} - f_{lP}^{Old} - \left.\frac{dF}{dT}\right|_{fl_P^m} \times (F^{-1}(f_{lP}^{m})) \right\}}_{Q_p} + \underbrace{\frac{-\delta H_P \times V_P}{\Delta t} \times \frac{dF}{dT}}_{R_P} \times T_P^{m+1} \tag{3-37}$$

One may notice now, how the discontinuities in Figure 3-1can cause difficulties in calculating the transient term of the energy equation.

Jump discontinuities in the $F(T)$ curve, Figure 3-1, are dealt with by setting $\left(\frac{dF}{dT}\right)_P$ to an arbitrarily large value. This will have the effect of forcing the solution to the phase change temperature as required in cells that are changing phase.

Equation (3-36), is used to update the value of the liquid fraction; however, the author of [66] found that for a smooth transition it is the best to be used only at the transition points from a fully-liquid phase to a phase change state where $f_l^p < 1$. The liquid-fraction correction of Voller [74], Equation (3-38), works best at nodes where $f_l^p = 1$. This gives:

$$f_{l_p}^{m+1} = f_{l_P}^m + \frac{a_p \left[T_P^{m+1} - F^{-1}(f_{l_P}^m)\right]}{V_P \delta H_P^{m+1} + a_P \times \frac{1}{(dF/dT)}} \tag{3-38}$$

where $\frac{dF}{dT}$ is evaluated at $f_{l_p} = f_{l_P}^m$. For nodes at the eutectic temperature, Equation (3-25) is used to update the value of f_l in the current time steps.

The only state that may potentially cause oscillations is when the $F(T)$ curve is shallow and as a result $\frac{dF}{dT} << 1$. In such a case, it is possible that $f_{l_p}^m = 0$, while T_P^{m+1} lies in the phase change region. To overcome this problem the direct update of Cross et al. [75] suggested by [61] is used in the current work and that is:

If $f_{l_p}^m = 0$ and

$$T_P^{m+1} \in \left[T_{liquidus}, T_{eutectic}\right] \tag{3-39}$$

Then

$$f_{l_P}^{m+1} = F(T_P^{m+1}) \tag{3-40}$$

For the convection source term, $S_{T_c} = \nabla.(\rho L_{fs}\vec{u})$, recall that the velocity is defined in the case of a mushy fluid by $\vec{u} = \vec{u}_l = \vec{u}_s$ and in the case of a columnar zone by $\vec{u} = f_l\vec{u}_l$, [11]. One may notice the differences between these two models more clearly by looking at the convective component of the source term in the energy equation of each model $S_{T_c} = \nabla.(\rho L_{fs}\vec{u})$ in the columnar zone model versus $S_{T_c} = \nabla.(\rho f_l L_{fs}\vec{u})$ in the mushy fluid model.

In the mushy fluid model the solid fraction is a function of space. The mass flow of liquid is thus not necessarily conserved across a given control volume, and the latent heat is convected with the flow since it is associated with the liquid phase. In the case of a columnar zone, however, the solid flow is zero, and no net latent heat is convected. This is more clear by noting that, due to the continuity equation, the convective term, $S_{T_c} = \nabla.(\rho L_{fs}\vec{u})$, is zero.

To retain the positive elements of the mushy fluid model and, on the other hand, keeping the assumption of the porous media type mushy zone, a "mixed model" was proposed and applied by Voller et al. [10] in which the convective source term in the energy equation is given by the one proposed in the mushy fluid model. In this way the required nature of the flow is achieved by specifying the Carman-Kozeny source term as well as retaining the elements of the mushy fluid model.

Now, the convective term in the source term of the energy equation can also be treated analogously to other convective terms in the governing equations including any higher order schemes such as QUICK or even Upwind Scheme. One must notice that in this case the variable is $f_l L_{fs}$ instead of usual variables such as temperature or velocity.

- *Diffusion terms*

The only remaining terms are the diffusion coefficients and fluxes at the CV faces in the momentum and energy equations. Consider a general diffusive balance for an independent variable continuous at the face between two CVs of the form:

$$\Gamma_1\left(\frac{\partial\phi}{\partial n}\right)_1 = \Gamma_2\left(\frac{\partial\phi}{\partial n}\right)_2 \tag{3-41}$$

where Γ_1 and Γ_2 are diffusion coefficients such as μ_{mix}, k_{mix}. Using one-sided estimates to the derivatives at the faces, the requirement of Equation (3.41) for the east face of volume P may be discretized as follows:

$$\Gamma_1\frac{\phi_e-\phi_P}{\Delta x_{Pe}} = \Gamma_2\frac{\phi_E-\phi_e}{\Delta x_{eE}} \tag{3-42}$$

An estimate for the face value of the dependent variable ϕ can be obtained in the form:

$$\phi_e = \left(1+\frac{\Gamma_2}{\Gamma_1}\frac{\Delta x_{Pe}}{\Delta x_{eE}}\right)^{-1}\phi_P + \left(1+\frac{\Gamma_1}{\Gamma_2}\frac{\Delta x_{eE}}{\Delta x_{Pe}}\right)^{-1}\phi_E \tag{3-43}$$

Inserting Equation (3-43) in either side of Equation (3-42), the estimate to the diffusive flux is given as:

$$\Gamma_1\left(\frac{\partial\phi}{\partial n}\right)_1 = \Gamma_2\left(\frac{\partial\phi}{\partial n}\right)_2 \approx \frac{\phi_E - \phi_P}{\frac{\Delta x_{Pe}}{\Gamma_1} + \frac{\Delta x_{eE}}{\Gamma_2}} \tag{3-44}$$

Note that Equation (3-44) is a local resistance analogy, or the harmonic mean formulation of Patankar [72].

3.3- Summary

To summarize, in the first section of this chapter the formulation used in the current work, including mass, x-y-momentum, and source-based energy transport equations, is explained. The volume fraction update is made in the mushy region by using Scheil's equation and a set of correlations.

In the second section, the discretized equation corresponding to the transport equations, based on the fully-implicit time integration, was presented. For each discretized transport equation the key points of implementation were outlined.

Next chapter introduces the VOF method and the solution methodology used in the current work.

Chapter 4

4. VOF technique and solution methodology

As it was explained in Chapter 2, the VOF method is one of the most popular methods in capturing interfaces. However, there are no sources in which the algorithm and the method are explained in detail in one paper or article. In this chapter, details of the VOF method applied in the current work are outlined. The basis of the method used in this work is mostly the one proposed by Pilliod and Puckett [48] and implemented by [50, 51, 53, and 76]. Nevertheless, in the current work, there are some parts of the algorithm, at which methods used by other authors, have been implemented.

To be able to explain the method of VOF, one has to make an initial guess for values of f in the domain. This can be done either at the beginning of the solution algorithm or where an update to the value of f is applied to the solution procedure. For example, in the current work, the update to the value of f is made right after updating the temperature field by one of the equations (3-36 to 3-40) implicitly and an initial guess for f distribution is made at the beginning of the solution's algorithm.

Knowing the value of f in a cell, a set of actions must be done to construct and reconstruct the interface in only mixed cells. In the following sections, 4.1-4.4 a detail of these procedures is explained. At the end of this chapter, section 4.5, a solution algorithm is presented.

4.1-Normal to the interface approximation and the identity of mixed cells

4.1.1-Normal to the interface

In vector calculus, the gradient of a scalar field is a vector field which points in the direction of the greatest rate of increase of the scalar field, and whose magnitude is the greatest rate of change.

In the original method of Youngs et al.[50] an approximation to ∇f is taken to point the direction normal to the approximate interface. ∇f can be calculated with the following difference scheme :

$$\begin{aligned} m_x &= \frac{\partial f}{\partial x} = \frac{f_E - f_W}{2} \\ m_y &= \frac{\partial f}{\partial y} = \frac{f_N - f_S}{2} \end{aligned} \tag{4-1}$$

The approximate values of f in the E, W, N and S control volumes, considering Figure 3-2, are calculated as bellow[50]:

$$\begin{aligned} f_E &= \frac{1}{2+\alpha}(f_{i+1,j-1} + \alpha f_{i+1,j} + f_{i+1,j+1}) \\ f_W &= \frac{1}{2+\alpha}(f_{i-1,j-1} + \alpha f_{i-1,j} + f_{i-1,j+1}) \\ f_N &= \frac{1}{2+\alpha}(f_{i-1,j+1} + \alpha f_{i,j+1} + f_{i+1,j+1}) \\ f_S &= \frac{1}{2+\alpha}(f_{i-1,j-1} + \alpha f_{i,j-1} + f_{i+1,j-1}) \end{aligned} \tag{4-2}$$

where α is a free parameter. Pilliod and Puckett [51] report that α = 2 will produce best results.

A modification of Youngs [50] idea has been used widely in recent works including [53]. This modification was made to somehow consider the effect of the different length and width of a computational cell in a non-uniform grid and also get rid of the free parameter α. The modified formulation to approximate the normal is written bellow:

$$m_x = \frac{1}{2}\times\frac{1}{4}\left(\begin{array}{l}\dfrac{f_{i,j-1}-f_{i-1,j-1}}{x_{i,j-1}-x_{i-1,j-1}}+\dfrac{f_{i,j}-f_{i-1,j}}{x_{i,j}-x_{i-1,j}}+\dfrac{f_{i,j}-f_{i-1,j}}{x_{i,j}-x_{i-1,j}}+\dfrac{f_{i,j+1}-f_{i-1,j+1}}{x_{i,j+1}-x_{i-1,j+1}}+ \\ \dfrac{f_{i+1,j-1}-f_{i,j-1}}{x_{i+1,j-1}-x_{i,j-1}}+\dfrac{f_{i+1,j}-f_{i,j}}{x_{i+1,j}-x_{i,j}}+\dfrac{f_{i+1,j}-f_{i,j}}{x_{i+1,j}-x_{i,j}}+\dfrac{f_{i+1,j+1}-f_{i,j+1}}{x_{i+1,j+1}-x_{i,j+1}}\end{array}\right)$$

(4-3)

$$m_y = \frac{1}{2}\times\frac{1}{4}\left(\begin{array}{l}\dfrac{f_{i-1,j}-f_{i-1,j-1}}{y_{i-1,j}-y_{i-1,j-1}}+\dfrac{f_{i,j}-f_{i,j-1}}{y_{i,j}-y_{i,j-1}}+\dfrac{f_{i-1,j+1}-f_{i-1,j}}{y_{i-1,j+1}-y_{i-1,j}}+\dfrac{f_{i,j+1}-f_{i,j}}{y_{i,j+1}-y_{i,j}}+ \\ \dfrac{f_{i,j}-f_{i,j-1}}{y_{i,j}-y_{i,j-1}}+\dfrac{f_{i+1,j}-f_{i+1,j-1}}{y_{i+1,j}-y_{i+1,j-1}}+\dfrac{f_{i,j+1}-f_{i,j}}{y_{i,j+1}-x_{i,j}}+\dfrac{f_{i+1,j+1}-f_{i+1,j}}{y_{i+1,j+1}-y_{i+1,j}}\end{array}\right)$$

(4-4)

Basically, the values of f initially approximated at the four corners of each mixed cell , i.e. North-East, North-West, South-East and South-West of each control volume, using values of f in neighboring cell's center, as shown in Figure 4-1,

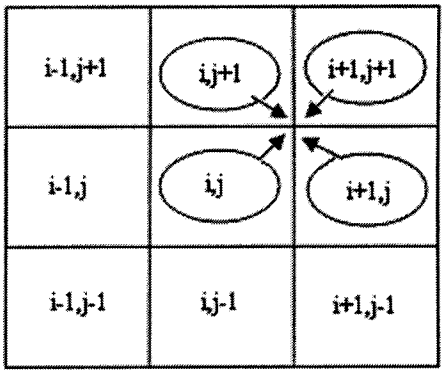

Figure 4-1, Normal approximation, step1

The final value of f at the cell center is the average value of these four corner values, see Figure 4-2.

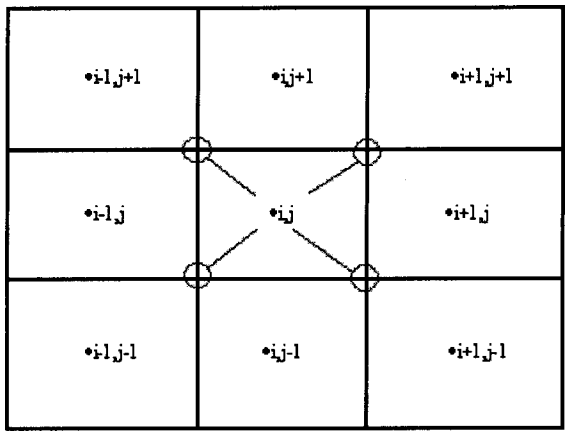

Figure 4-2, Final normal approximation

Hence n_x and n_y components of the unit normal vector n are

$$n_x = -\frac{m_x}{\sqrt{m_x^2 + m_y^2}}$$
$$n_y = -\frac{m_y}{\sqrt{m_x^2 + m_y^2}} \qquad (4\text{-}5)$$

In the current work, the normal vector in a mixed cell always points to the partial solid region of the cell. Once the normalized vector $\vec{n}$ is calculated, a straight line can be positioned perpendicular to it in such a way that, it matches with the value of f. Pilliod and Puckett [48] show that this method does not construct all linear interfaces exactly, and is at best first-order accurate. In addition, to have a higher order fit one needs to use a reconstruction algorithm. For this reason, in the present work, this method is used only as an initial guess for the interface reconstruction algorithm and, instead the LVIRA (Least square Volume of Fluid Interface Reconstruction Algorithm) scheme is used to determine final approximation of the normal to the interface $\vec{n}$.

4.1.2- mixed cell's identity

Knowing an estimate of the normal can facilitate information to flag each computational cell based on the direction of the normal to the interface. In the current work the flagging method of Shirani et al. [53] is used to flag each mixed cell.

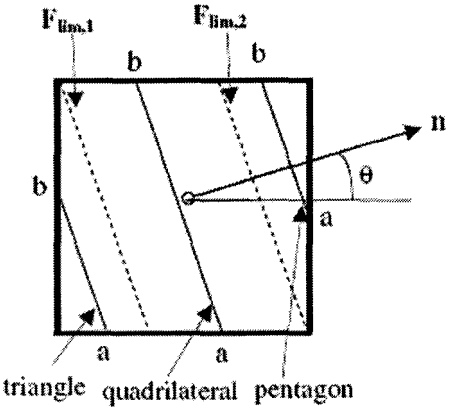

Figure 4-3, Cell identity [53]

Depending upon the position of the interface and the value of f in a mixed cell, Figure 4-4 shows 24 possible mixed cells out of 26 possible cases. The other cases exist

when the interface is parallel to x-y axes. The normal vector angle, θ, can take any value between zero and 2π ,and create any of the 24 cases shown in Figure 4-4. Figures 4-3 and 4-4 show that the area filled with fluid can be triangle, quadrilateral or pentagon, depending on the value of f, n_x and n_y in that cell. The limiting value of f , see Figure 4-3, for the cases when n is in the first octant and $n_x > n_y$ are:

$$f_{\text{lim},1} = \frac{n_y}{2n_x} \qquad f_{\text{lim},2} = 1 - f_{\text{lim},1} \tag{4-6}$$

and for a general case where n is arbitrary are:

$$f_{\text{lim},1} = \frac{n_{\min}}{2n_{\max}} \qquad f_{\text{lim},2} = 1 - f_{\text{lim},1} \tag{4-7}$$

where

$$\begin{aligned} n_{\min} &= \min\left(|n_x|, |n_y|\right) \\ n_{\max} &= \max\left(|n_x|, |n_y|\right) \end{aligned} \tag{4-8}$$

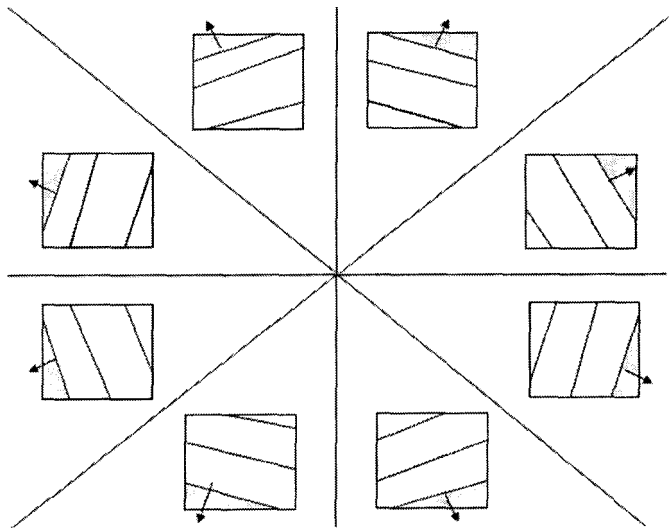

Figure 4-4, Different interface configuration

Base upon the magnitude of normal vectors in x and y direction and their values, the location of the cell in each octant of Figure 4-4 can be found and depending on the values of $f_{\text{lim},1}$ and $f_{\text{lim},2}$, different possibilities of having a triangle, pentagon or quadrilateral can be obtained.

4.2- Interface intercept or line constant

Knowing the normal to the interface gives the angle of the interface. To identify the exact location of the interface in a 2D simulation, the interface line intercept is necessary. Determining the constant of the line is the most difficult task in the reconstruction algorithm. The constant must be calculated in such a way that the resulting line passing through the origin of the system of coordinates, passes through the cell with a truncation volume equal to the cell's material volume, i,e. fluid volume in the mixed cell [23].

This task can be completed by creating a Volume (distance) or a V (d) relation, in which V can vary linearly, quadratically, or cubically with 'distance'. This relation

depends on the coordinate system and the shape of the truncated control volume formed by the interface segment.

Rider and Kothe, [23], attempted to construct an algorithm for determining the constant based on the direct solution of the V (d) relation. However, since this relation is often non-linear and varies in each mixed cell, the method was not efficient. Instead, Rider and Kothe [23] chose to create the V (d) relation iteratively in each mixed cell. Using this method, the resulting algorithm is independent of the mixed cell's properties and data.

The distance, hereafter called d, can be found when the nonlinear Function (4-9) is satisfied; i.e.:

$$f(d) = V(d) - V = 0 \tag{4-9}$$

V (d) is the volume of the constituent in the cell bounded by the interface segment and the portion of the mixed cell edges within the constituent.

When these two volumes are equal, (to within some tolerance, 1.0E-4), it means that the interface is declared and the reconstruction task is over, so the values of vector n and scalar d are correct.

One can also calculate the distance between the interface and the cell center; offset length, L, instead of calculating d, [77, 78, and 36]. This approach recently has been used widely and is used in the present work. Basically, given the value of f from either volume fraction evolution equation, Equation (3-36) or (3-38) to (3-40), or an initial guess, and approximated normal to the interface, the task is finding the optimum value of L by finding the root of equation:

$$(f\_new - f\_given) = 0 \tag{4-10}$$

f_new is the resulted value of f base on the first guess for value of L. In other words, starting with a guess for L, the task is minimizing the error between the given value of f, f_given, and the resulting f, f_new.

A host of root finding algorithms is available to determine the roots of any function [79]; herein, a Secant method is chosen to find the zero of Function (4-10) [77]. The Secant method requires less computational effort in comparison with Newton's method and other similar methods because an evaluation of the derivative of V with respect to L is not required [77]. Hence, the Secant method needs two initial guesses. A good initial guess can increase the rate of convergence. The maximum possible distance of an interface from a mixed cell center with a dimension of $\Delta x \times \Delta y$ can be calculated easily;

$$\begin{aligned} L_1 &= \left|\vec{n}_x.\Delta x + \vec{n}_y.\Delta y\right| \\ L_2 &= \left|-\vec{n}_x.\Delta x + \vec{n}_y.\Delta y\right| \\ \text{Maximum L} &= 0.5 \times Max(L_1, L_2) \end{aligned} \tag{4-11}$$

Using a small index of this Maximum could be a good candidate for the first initial guess in the Secant method. The Second initial guess could be obtained by subtracting the initial value from the maximum L.

To find out what the value of f_new corresponding to the value of L is, one needs to calculate the volume of constituents in each mixed cell. Obviously, to be able to

calculate the volume of fluid in each mixed cell, one needs to construct an interface segment in each mixed cell.

4.3- Line segment in mixed cells

To construct a line segment, one needs to calculate the endpoints of the interface in a mixed cell. The general idea of line segment construction used in this work comes from reference [23], however the details of the idea have not been presented completely in any articles or journals. Details are presented below.

4.3.1- Normal location on the interface and interface endpoints on the faces

Knowing the estimate of normal, approximated from either Youngs method, explained previously, or LVIRA scheme, will be explained later, and the value of L, the coordinates of the base point of the normal can be calculated in such a way that

$$X_{base} = X_{center} - L \times n_x \tag{4-12a}$$

$$Y_{base} = Y_{center} - L \times n_y \tag{4-12b}$$

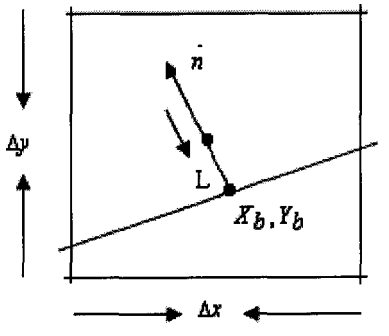

Figure 4-5, Line segment construction, X_b, Y_b are the base point's coordinates

This point is taken as a point on the interface to help finding the end points of the interface. For storing purposes, one array that takes maximum two values is set to store the endpoints of the intersection, i.e.

$$X_{\text{int}erface}(1), Y_{\text{interface}}(1)$$
$$X_{\text{int}erface}(2), Y_{\text{inter}face}(2)$$

Each face of a mixed cell is also numbered to identify which face intersects with the interface, i.e. "1" for the bottom face, "2" for the right face, "3" for the top and "4" for the left face of each mixed cell. So if, for example, $n_y \approx 0.0$, assuming an interface parallel to the x- axis, and $Y_{bottom} \le Y_{base} \le Y_{top}$, the two intersection points are:

$$X_{\text{int}erface}(1) = X_{right\ face}$$

$$Y_{\text{int}erface}(1) = Y_{base}$$

$$face\_number(1) = 2$$

,and

$$X_{\text{int}erface}(2) = X_{left\ face}$$

$$Y_{\text{int}erface}(2) = Y_{base}$$

$$face\_number(2) = 4$$

The same procedure can be done for an interface parallel to the Y-axis.

For interfaces not parallel to the x-y axis, for sake of simplicity, one can define two different slopes, one for cases that intersect the top and bottom faces, $m = \dfrac{-n_y}{n_x}$, and

one for cases that intersect right and left face, $m' = \frac{-n_x}{n_y}$. Then, by setting test points on each faces, interface points can be found easily. To clarify the above discussion, assume there is a mixed cell in the domain with an interface that is not parallel with the X-Y axis, and intersects with the top and bottom faces of the cell. A test point can be defined as:

$$\begin{cases} Y_{test} = Y_{top-face} \\ X_{test} = m \times (Y_{test} - Y_{base}) + X_{base} \end{cases} \tag{4-13}$$

If $X_{left-face} \le X_{test} \le X_{right-face}$ then,

$$\begin{cases} X_{\text{int}} = X_{test} \\ Y_{\text{int}} = Y_{test} \end{cases} \tag{4-14}$$

The same test can be done for the other faces [23].

4.3.2-Polygon collection

Knowing the interface segment in a mixed cell leads to an examination of the vertices of the cell to see if they are located in the constituent of interest. A polygon can be constructed to calculate the volume of a constituent bounded by the faces of the mixed cell and the interface line.

To fulfill this task, test points are chosen to validate the region. For example, to check the lower left corner, the test point can be the base point, $\begin{cases} X_{test} = X_{base} \\ Y_{test} = Y_{base} \end{cases}$ and if

$$(X_{left} - X_{test}) \times n_x + (Y_{bottom} - Y_{test}) \times n_y \ge 0$$

then the vertex is in the constituent. The next step will be checking the bottom face to see if it is within the constituent and so on. It is worth mentioning that this task is done in

counterclockwise order to be consistent with the line segment finding task. Having the vertices of a polygon, the area of the polygon can be calculated using:

$$A = \frac{1}{2}\sum_{v=1}^{n}(x_v y_{v+1} - x_{v+1} y_v) \tag{4-15}$$

where, (x_v, y_v) is the vertex of an n-sided polygon, collected in the counterclockwise order.

By constructing the polygon, enough information is available to find the value of f_new.

4.4-LVIRA (Least Square Volume of Fluid Interface Reconstruction Algorithm)

Having the estimated value of **n** from Youngs method and the offset length, gives us a first order approximation of the interface in each mixed cell. LVIRA was developed by Puckett [36] and is a second order accurate reconstruction method to reproduce linear interfaces. The advantage of the LVIRA method over the PLIC and similar reconstruction methods is in reproducing continuous interfaces across the cell boundaries.

The concept of the LVIRA scheme can be understood as a function which returns **n**, the orientation of the interface and L, the offset distance of interface from the cell center, given the volume fraction distribution of the complete domain.

In the LVIRA method the following G function

$$G_{i,j}(n, L) = \sum_{k,l=1}^{3}\left\{f_{i+k,j+l} - \tilde{f}(n, L)_{i+k,j+l}\right\}^2 \tag{4-16}$$

is minimized for a block of 3×3 computational cells to approximate the interface in the cell center (i, j) subject to the constraint at the cell center of 3×3 block of cells $f_{i,j} = \tilde{f}(n, L)_{i,j}$ (Tolerance=1.0E-8). $f_{i,j}$ is the actual volume fraction of the cell (i, j) and $\tilde{f}(n, L)_{i,j}$ is the function that plots the line with orientation **n** and offset length L into the cell (i, j).

L is obtained by considering the constraint $f_{i,j} = \tilde{f}(n, L)_{i,j}$, and $G_{ij}(\vec{n}, L)$ is basically the squared deviation of the actual volume fraction from the volume fraction obtained by drawing the interface using the same line for the entire 3×3 block of cells centered at cell (i, j).

Minimization of this function rotates and translates the line in such a way that ensures the drawing is exact for the center cell and in such a way that the straight line associated with this drawing is the best fit to the volume fractions for the eight surrounding cells [77]. The outcome of the whole minimization process is the calculation of the unit normal and the offset length (**n**, L) for the cell at the center of the 3×3 block. This calculation is performed for all the mixed cells, satisfying the constraints that the drawing $\tilde{f}(n, L)_{i,j}$ returns the actual volume fraction [77].

4.5- Solution Procedure

4.5.1- Interface Reconstruction algorithm

Knowing the details of the reconstruction of the interface in a mixed cell, the VOF reconstruction algorithm can be summarized as below:

1) Given the distribution of f; only for cells that have value of f between zero and one, approximate a normal to the interface and identify the mixed cell

2) Find the distance between the interface and the cell center.
3) Define a line segment and truncate the mixed cell with this line segment.
4) Find and assemble the N vertices of the polygon formed by those cell vertices inside the fluid and interface line/cell edges intersection points.
5) Compute the volume bounded by this polygon.
6) Determine if the polygon volume differs from the known fluid volume by some prescribed tolerance.
7) If the volume differs, find a new estimate for L and go back to step 3.
8) If the volumes do not differ, save the values of n and L in all mixed cells and calculate the angle that the interface makes with the x-axes.
9) Start LVIRA for the best fit of the interface in the center cell of a block of 3 by 3 by rotating the interface with a small angle in positive and negative direction and minimizing equation (4-16).
10) Check if $f_{i,j} = \tilde{f}(n,L)_{i,j}$ with some prescribed tolerance. If it is satisfied then the interface is declared reconstructed. If it is not, rotate the interface until the best fit results. The whole algorithm can be shown in a more compact form as shown in Figure 4-6

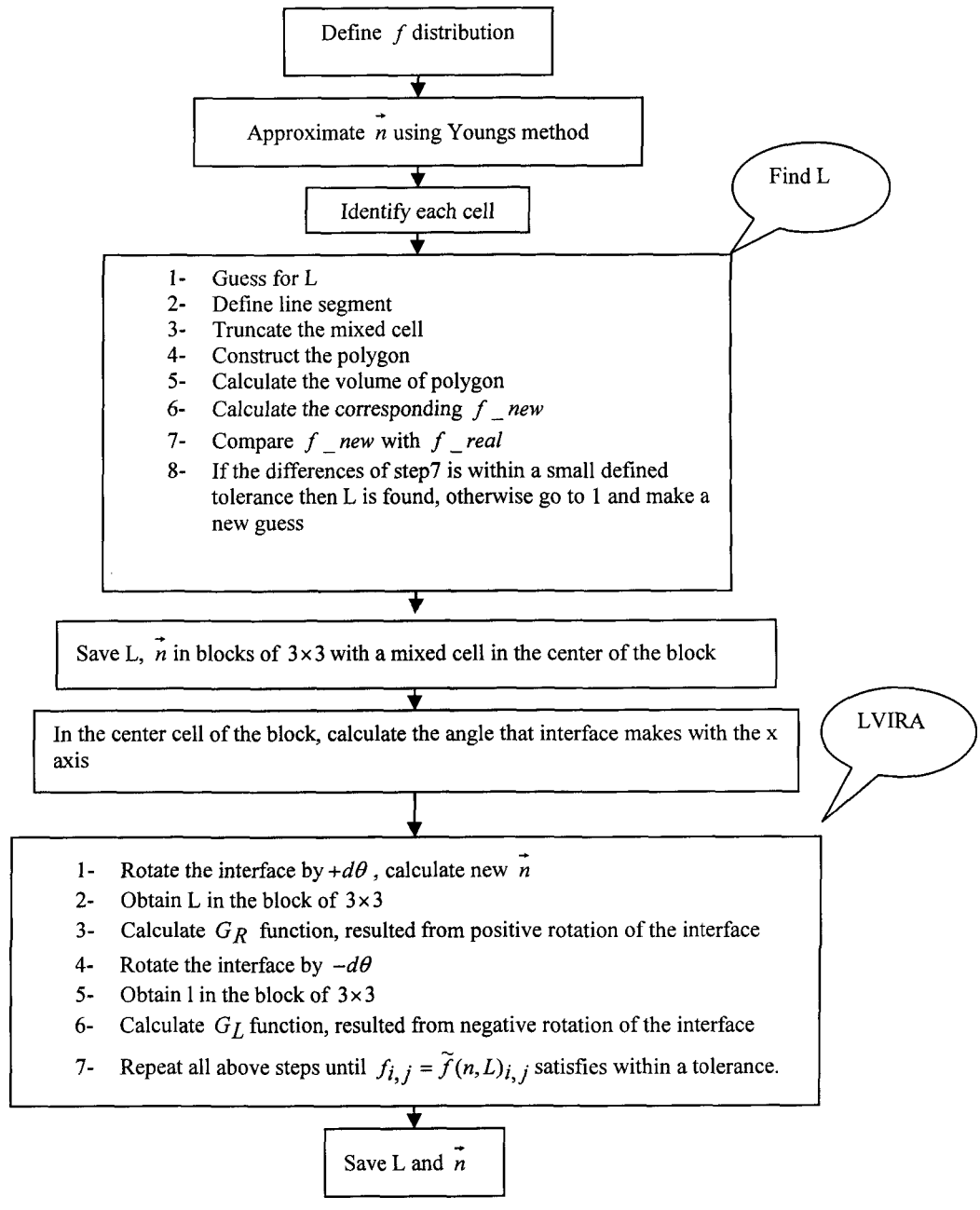

Figure 4-6, Flow chart of VOF algorithm

4.5.2- Solution procedure

To solve a complete solidification problem the following algorithm, which consists of three main elements, is chosen:

1- Pre-processing:

- Domain definition
- Grid generation
- Set initial field variables $(u, v, T, \hat{u}, \hat{v})$.
- Guess an initial distribution of f in the domain.

2- Processing and solving:

- Approximate normal to the interface using Youngs method and identify mixed cells
- Calculate Γ, will be explained in Chapter 5, to be able to approximate pressure gradient in the vicinity of the interface
- Update physical properties including ρ and μ based on the value of f.
- Compute pressure gradient based on the initial values
- Initialize residual
- **Begin the time-step loop (outer loop)**
- Set old values of the field variables equal to the new values.
- **Begin the linearization loop (inner loop)**
- Update all values of normal, Γ as well as cell identity based on the updated value of f
- Update physical properties including ρ and μ based on the value of f.
- Compute active coefficients in Energy and mass-momentum equations
- Compute all source terms
- Read all the boundary conditions
- Add the pressure terms to the momentum equations

- Check the residual against convergence criteria
- Solve descretized energy equation using WATSOL1 in WATSIT-B solver
- Update f using equations (3-36 to 3-40)
- Call the interface reconstruction template.
- Solve descretized coupled mass-momentum equation
- Update pressure gradient and advecting velocities
- **End Linearization loop**
- **End the time-step loop**

3- Post-processing:

- Domain geometry and grid display
- Vector and contour plots

The matrix solver WATSIT-B includes two subroutines watsol1.f and watsoln.f. They are iterative sparse matrix solvers that operate on the system

$$Ax = b$$

where A is a banded $N \times N$ sparse, non-singular matrix, and b and x are N-length vectors. The vector b is the source term (or right-hand side), and x is the solution vector [80]. Watsol1.f is created to operate on sparse matrices with scalars as the elements. In the current work this package is used to solve energy equation.

Watsoln.f is designed to take advantage of block structure in matrices and it is useful to solve a system of equations where several variables are being solved at each node at the same time. This package is used to solve the coupled mass momentum equation in the current work where pressure, x and y direction velocities are unknown

variables. The sparse matrix storage scheme in this solver is the one used in the Yale Sparse Matrix Package. One interested in details can find more information in [80].

4.5.3- Convergence criterion

Determining when to stop the iterative process is not a trivial problem. Depending on the size of the problem, each time step could consist of several nonlinear iterations. In the current work, the solution convergence is checked through normalized residuals. To make it clear, consider a discrete equation for a field variable φ. For each coefficient update in the linearization loop, the current values of the field variables are used to construct coefficients (Equation 4-17). Then, the discrete equations for the previous time step are solved (Equation 4-18).

$$a_P^{m-1}\varphi_P^m = a_W^{m-1}\varphi_W^m + a_E^{m-1}\varphi_E^m + a_N^{m-1}\varphi_N^m + a_S^{m-1}\varphi_S^m + b_P^{m-1} \tag{4-17}$$

$$a_P^m\varphi_P^m = a_W^m\varphi_W^m + a_E^m\varphi_E^m + a_N^m\varphi_N^m + a_S^m\varphi_S^m + b_P^m \tag{4-18}$$

After each coefficient update, the residual of the discrete equations can be easily checked:

$$R_P^m = a_P^m\varphi_P^m - a_W^m\varphi_W^m - a_E^m\varphi_E^m - a_N^m\varphi_N^m - a_S^m\varphi_S^m - b_P^m \tag{4-19}$$

The absolute value of R_P^m is the local residual. To get an indication of the convergence behavior across the whole flow field, a global residual can be defined as the maximum of the local residuals over all n control volumes. Then the size of this global residual is a reference of how closely the non-linear problem has been solved. The global residual will have a larger value in simulations where the flow variable φ has a larger

magnitude, so a different truncation value is necessary for R^m. A normalized global residual can be defined to remove the effect of the magnitude of φ. In the current work the normalizing factor is defined as:

$$\frac{|\varphi_{new} - \varphi_{old}|}{\left(\frac{\sum_i |\varphi_i|}{n}\right)} \tag{4-20}$$

Convergence of the linearization loop in the current work is estimated to have occurred when the normalized residuals for the discrete, linearized governing equations, in a solidification problem, were reduced to below the value 10^{-4} for each control volume.

4.6- Summary

In this chapter, the method of VOF used in the present work was explained in detail. Also, a solution procedure for a solidification problem was outlined. Having all necessary information, in previous chapters, the author will validate the proposed formulation in the remaining chapters by solving a set of problems.

Chapter 5

5. Validations and Discussion

Three case studies were considered to rigorously validate all components of the proposed computational model. The first was a coupled mass-momentum problem solved for a stationary liquid-solid interface set by manually defining the values of f in the domain. The problem was then solved using the complete VOF code to demonstrate that the correct velocity and pressure fields are predicted adjacent to the interface.

The second case considered was a simple heat transfer problem in which the wall temperatures were set to be above the eutectic temperature. This case was also solved using the complete VOF code with the energy equation invoked and the result shows correctly that no solidification takes place and a simple thermal current is developed in the domain due to the differential wall temperatures.

The final case study includes three problems. The first problem considered is an Al-4.5%Cu cavity which has the left wall of an initially liquid domain set below the eutectic temperature, while the right wall is above the eutectic temperature and the upper and lower walls are insulated. In this case, the solution is a slowly moving solidification front that starts at the left wall and moves towards the right. The effect of convection in the mushy region is examined. Also, in this problem the link between the energy transport equation and VOF algorithm is validated. The second problem, which is for a quantitive comparison of this work to the validated work of Swaminathan and Voller [65], is a simplified solidification problem, i.e. the conduction dominant solidification problem of [65] without considering the convection effect in the mushy region. Finally,

the third problem is a thermal cavity filled with Al-6%Mg, with physical conditions similar to the first problem, but without convection terms in the energy transport equation. Further discussions will be presented in the following sections.

5.1- Case study 1: Virtual stationary interface problem

In this problem, only the velocity vectors and pressure are of interest. More particularly, attention is focused on obtaining an accurate pressure gradient in the momentum equation in the vicinity of the solid-liquid interface. A careful treatment of pressure gradient adjacent to the interface is necessary to remove the possibility of spurious currents. To study this, a 0.5 by 0.1 (m) channel is chosen as the domain of the problem. A relatively weak flow, with Reynolds number of 2 is imposed at the inlet of the channel. A pre-defined interface which divides the domain into solid and liquid regions is imposed in the channel by manually defining a distribution of f to create a channel with the shape illustrated in Figure 5-1A.

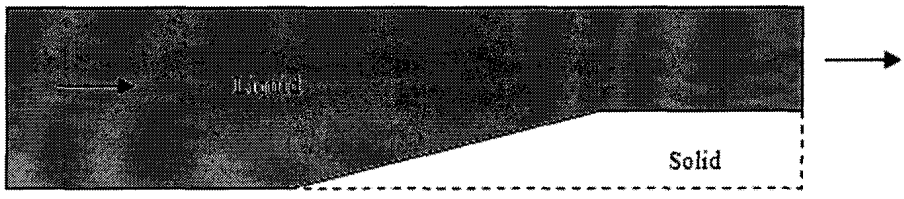

Figure 5-1A, Schematic showing the expected solid-liquid interface

By manually defining a distribution of f equal to 0.25 and 0.75 and 0.1 for chosen cells, a solid-liquid interface is defined at almost two-channel width from the entrance where the velocity profile has already obtained a parabolic shape. At this point, the VOF package with pre-defined values of f was applied to check the agreement of the

interface. It was found that the resulting interface was in agreement with the one shown in Figure 5-1A, see Figure 5-1B.

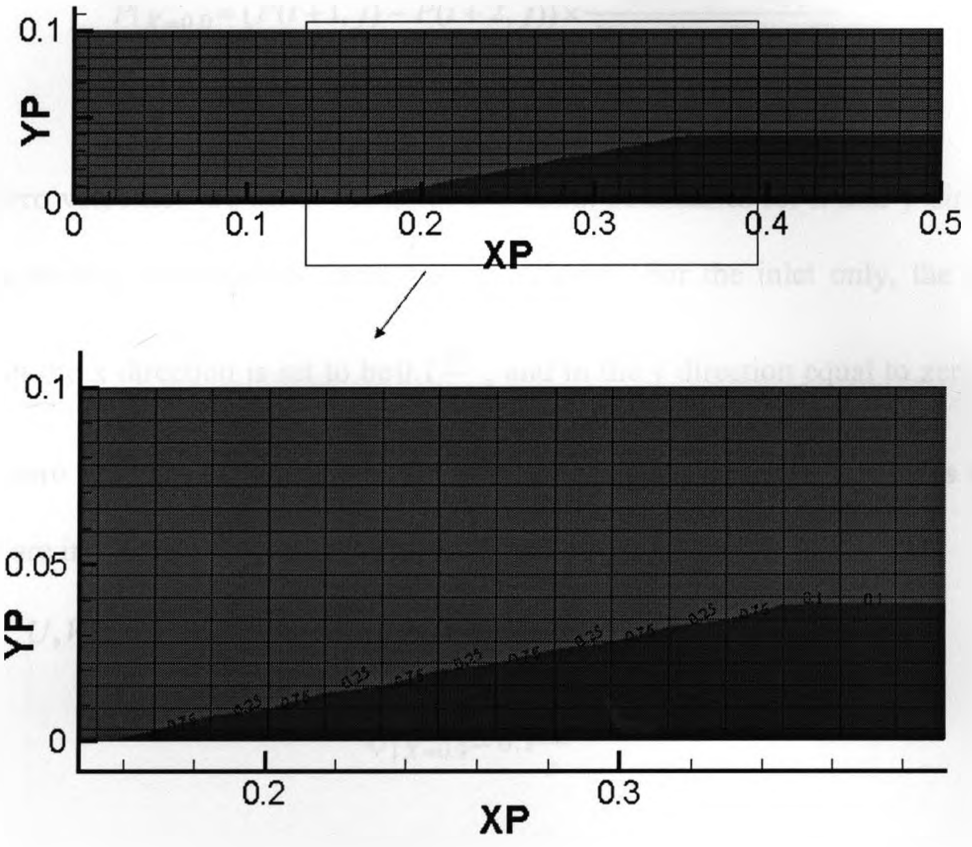

Figure 5-1B, Plot showing the solid-liquid interface in the domain by manually defining a distribution of f and using the VOF method

Boundary conditions for pressure and velocity are defined on the channel edges as below:

Pressures are set to be zero at the outlet (east face) and extrapolated at the north face, south face and west face of the channel:

$$P\big|_{X=0.5} = 0.0 \tag{5-1a}$$

$$P\Big|_{Y=0.0} = (P(i,j+1) - P(i,j+2)) \times \frac{Y(j+1) - Y(j)}{Y(j+2) - Y(j+1)} \tag{5-1b}$$

$$P\Big|_{Y=0.1} = (P(i,j-1) - P(i,j-2)) \times \frac{Y(j) - Y(j-1)}{Y(j-1) - Y(j-2)} \tag{5-1c}$$

$$P\Big|_{X=0.0} = (P(i+1,j) - P(i+2,j)) \times \frac{X(i+1) - X(i)}{X(i+1) - X(i+2)} \tag{5-1d}$$

Zero velocities are set at the north and south boundaries for x and y direction to enforce a no-slip impermeable condition on velocity. For the inlet only, the value of velocity in the x direction is set to be $0.1\frac{m}{s}$, and in the y direction equal to zero. At the outlet, a zero gradient of velocity in the x direction is set and zero velocities in the y direction are implemented to mimic a typical fluid outlet; i.e.:

$$U,V\big|_{X=0.5} = 0.0, \quad U,V\big|_{Y=0.1} = 0.0, \quad U,V\big|_{Y=0.0} = 0.0, \quad \mathrm{V}\big|_{\mathrm{X}=0.0} = 0.0$$
$$U\big|_{X=0.0} = 0.1\frac{m}{s} \tag{5-2}$$

Having the boundary conditions and values of f defined, the coupled mass-momentum equations were solved.

Figure 5-1C illustrates the pressure-velocity field at the area adjacent to the interface.

Such a result was expected as the typical approximation used for the pressure gradient in pure fluid flows may not be terribly accurate due to the presence of solid and fluid in a single cell. This task is complicated by the fact that the advected velocity changes dramatically in the vicinity of the interface due to the flow restriction. As a result, a jump in dynamic pressure occurs. As it is shown in Figure 5-1C , the velocity

vectors are directed into the solid region resulting in the presence of a current similar to the "Spurious Currents" reported in [1] in the vicinity of the interface. Using the VOF method, the interface only exists numerically based on the definition of f . As such, a boundary condition can not be applied at the interface to correct the velocity field.

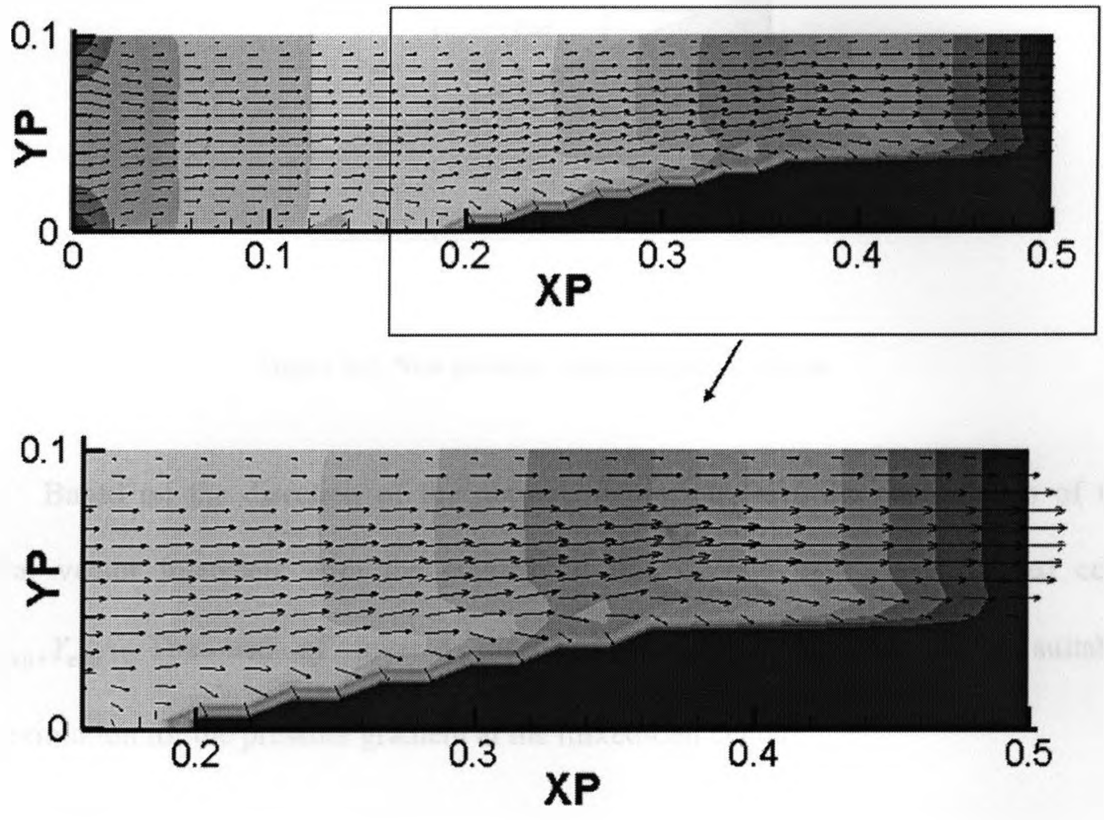

Figure 5-1C, Plot showing the Pressure and velocity field with spurious current in the vicinity of interface

To correct the spurious activity that results from inconsistent modeling of the pressure, a modification must be made to smooth the effect of the interface on the flow. A new approximation to the average pressure gradient terms appearing in momentum equation for cells that include an interface (mixed cell) is developed in this work to overcome this issue.

To obtain a suitable estimate for the pressure, consider the following sub-block of the domain, Figure 5-2, with a mixed cell surrounded by eight other control volumes.

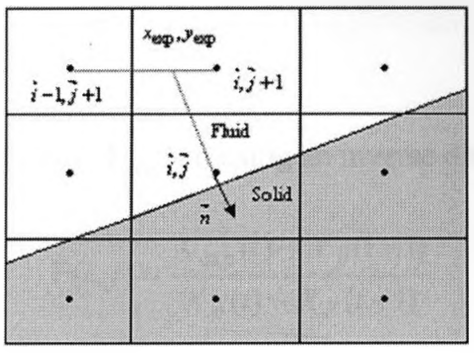

Figure 5-2, New pressure approximation scheme

Based on the direction of the normal to the interface, the continuation of the normal vector intersects with the connecting line of one pair of surrounded cells at X_{exp}, Y_{exp}. The use of inverse distance approximation gives us a suitable approximation for the pressure gradient at the mixed-cell center:

Knowing that

$$\begin{cases} Y_{exp}(j) = Y_p(j) + \beta n_y \\ X_{exp}(i) = X_p(i) + \beta n_x \end{cases} \tag{5-3}$$

Start with setting

$$Y_{exp}(j) = Y_P(j+1) \tag{5-4}$$

These result in

$$\beta = \frac{Y_{\exp}(j) - Y_P(j)}{n_y} \tag{5-5}$$

So,

$$X_{\exp}(i) = \beta n_x + X_P(i) \tag{5-6}$$

Now, Based on the value of $X_{\exp}, Y_{\exp}$ and using an inverse distance approximation,

$$\Gamma(i,j) = \frac{X_{\exp}(i) - X_p(i-1)}{X_p(i) - X_P(i-1)} \tag{5-7}$$

$$\left.\frac{\partial P}{\partial x}\right|_{i,j}^{ID} = \left(1 - \Gamma(i,j)\right) \times \left.\frac{dP}{dx}\right|_{i,j+1} + \left(\Gamma(i,j)\right) \times \left.\frac{dP}{dx}\right|_{i-1,j+1} \tag{5-8}$$

After applying this approximation into x-momentum equation, a similar approximation was made in the y-momentum, and the resulting pressure, f and velocity fields are shown in Figure 5-3.

One might notice that how a correct estimate of normal to the interface can affect the approximation.

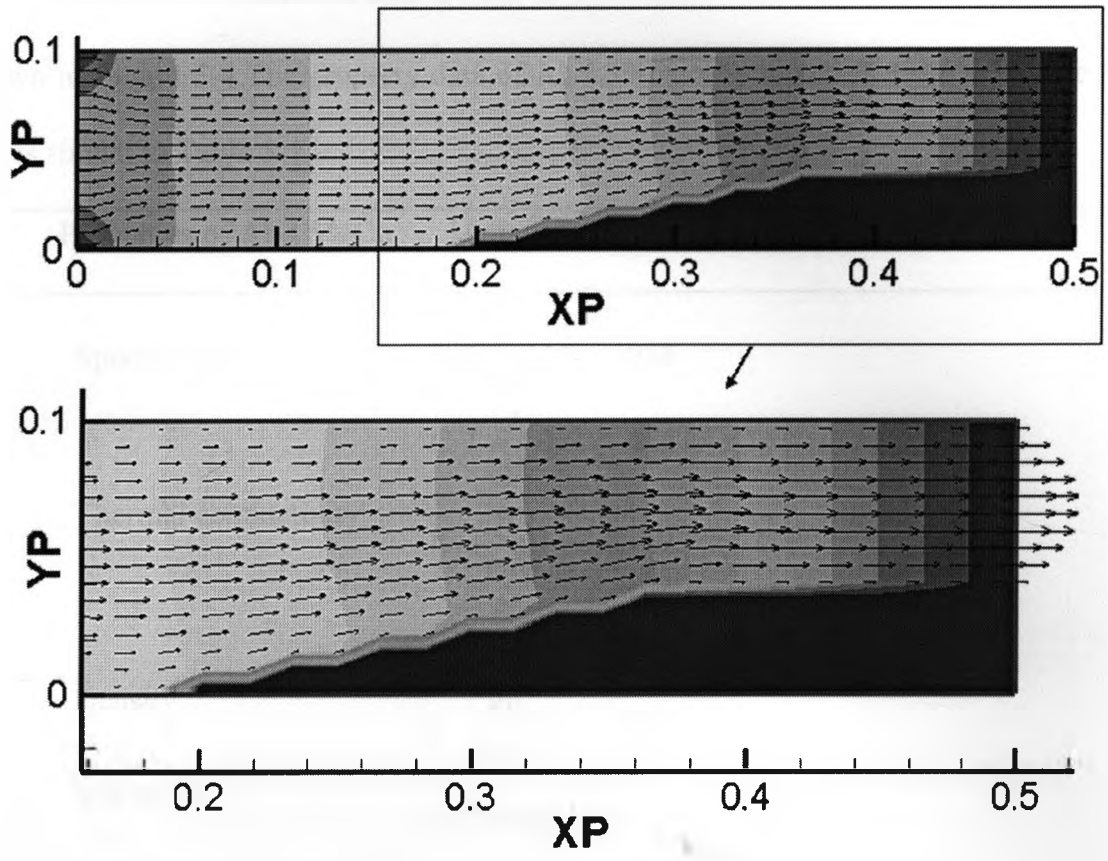

Figure 5-3, Plot showing corrected pressure and velocity field

5.2- Case study 2: Source based Energy equation problem

In this case, the source based energy equation formulated in Chapter3 is invoked and quantitively validated. Buoyancy-driven convection in a sealed cavity with vertical sides that are cooled and heated is a prototype of many industrial applications. The problem being considered is a two-dimensional flow and heat transfer in an upright square cavity of side 0.05 (m) filled with an Al-4.5% Cu binary alloy. The two horizontal walls are insulated and the vertical sides are at constant temperatures.

The hot wall and cold wall are set to $T_h = 933K$ and $T_C = 913K$, respectively, as shown in Figure 5-4.The property data of Al-4.5%Cu was retrieved from reference [81] and is shown in Table 5-1.

Properties of Al-4.5%Cu	Units	Value
Specific heat	$C_s\ [\frac{J}{kgK}]$	958
	$C_s\ [\frac{J}{kgK}]$	958
Thermal conductivity	$K_s\ [\frac{W}{mK}]$	180
	$K_l\ [\frac{W}{mK}]$	95
Density	$\rho\ [\frac{\text{kg}}{\text{m}^3}]$	2460
Viscosity	μ [Kg/m s]	1.17×10^{-4}
Latent heat of fusion	L [J/kg]	3.9×10^{5}
Eutectic temperature	T_E [K]	821.5
Melting temperature	T_m [K]	933.5
Liquidus temperature	T_l [K]	913
Thermal expansion	$\beta[1/K]$	2.5×10^{-6}

Table 5-1, Aluminum-Copper Data Properties of binary Al-4.5% Cu Alloy

The solution of the governing equations can lead to a complete understanding of the velocity and temperature fields for natural convection in a square cavity .

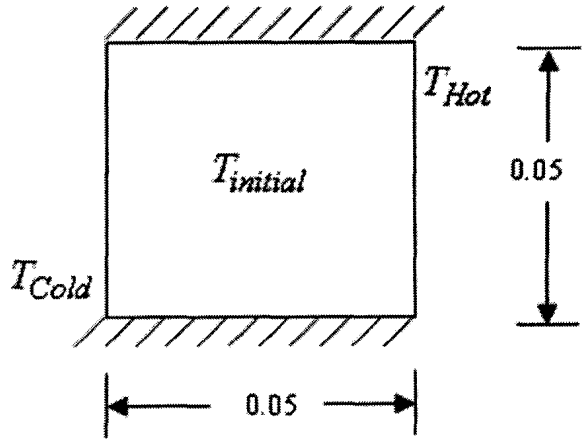

Figure 5-4, The differentially heated cavity used in the second case study

A two-dimensional non-uniform co-located grid system (40×40), shown in Figure 5-5, was used. Note that the grid is condensed toward the walls where gradients in velocity and temperature were expected to be highest.

Results

It is generally known that fluid flow in a buoyancy-driven convection system is characterized by a single large re-circulating cell. The lighter fluid rises along the hot wall. The fluid then descends along the cold wall as it gets heavier.

As it can be observed from the plot shown in Figure 5-6, the fluid that is heated next to the hot wall (right), rises and replaces the cooled fluid next to the cold wall (left), which drops, thus giving rise to a CCW rotating vortex.

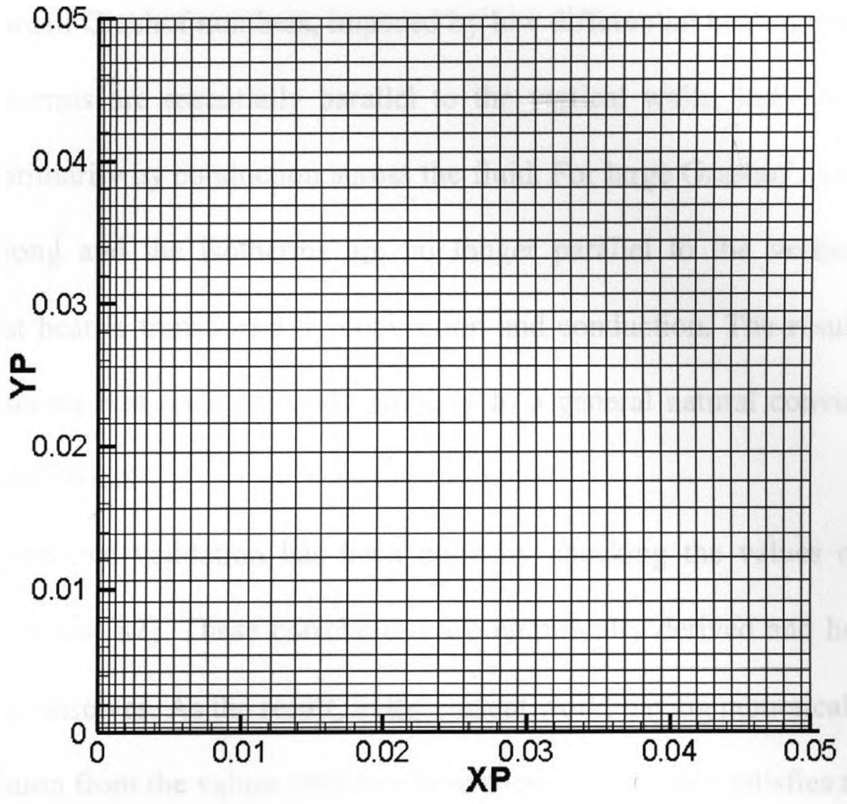

Figure 5-5, Plot showing the non-uniform grids used in case study-2

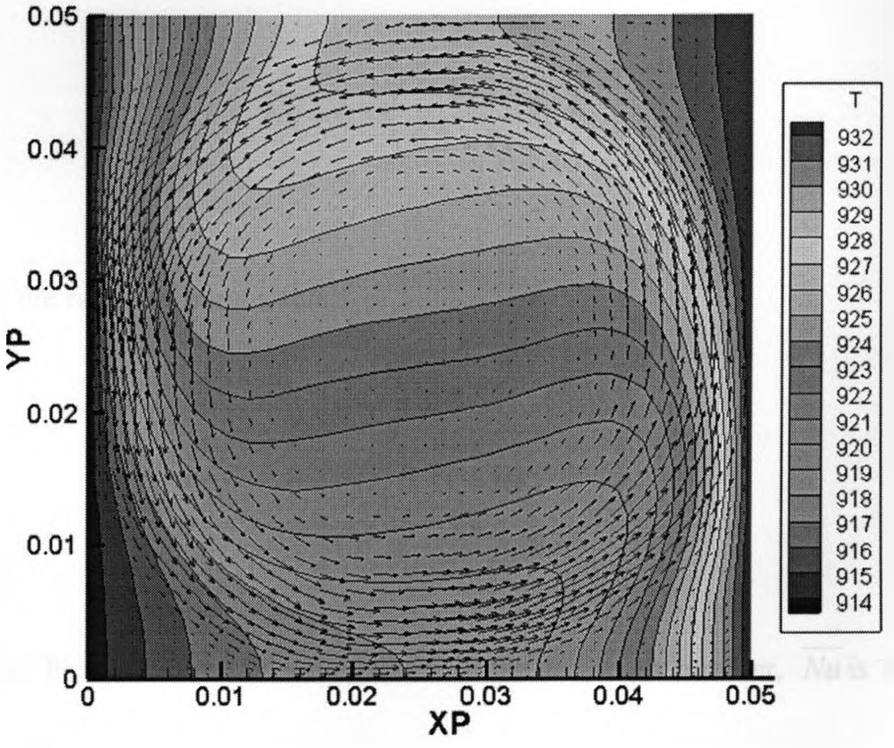

Figure 5-6, Plot showing temperature contours and velocity vectors for a differentially heated cavity

For small Grashof numbers, imposed by low differential heating, the flow is weak and the isotherms are essentially parallel to the vertical walls, indicating that heat is transferred primarily by conduction across the fluid. For large Grashof numbers, the flow becomes strong and the isotherms are no longer parallel to the vertical walls. This indicates that heat is transferred by convection and conduction. The result presented in Figure 5-6 shows that code responds properly to a general natural convection problem without solidification.

A quantitive validation has been done by checking the values of correlations given in reference [82]. These correlations are empirically derived and hold a range of uncertainty themselves. As the result, in the current work, having numerical values within 0-15% deviation from the values obtained from these correlations satisfies the author. For aspect ratios (height divided by length) close to one, following correlations are suggested in [82]:

$$\overline{Nu} = 0.18\left(\frac{\mathrm{Pr}}{0.2+\mathrm{Pr}}Ra\right)^{0.29} \tag{5-9}$$

Over the range of parameters:

$$\begin{cases} 1 \le \dfrac{H}{L} \le 2 \\ 10^{-3} \le \mathrm{Pr} \le 10^{5} \\ 10^{3} \le \dfrac{\mathrm{Pr}}{0.2+\mathrm{Pr}}Ra \end{cases} \tag{5-10}$$

where, Pr is the Prandtl number, Ra is the Rayleigh number, $\overline{Nu}$ is the Nusselt number and $\frac{H}{L}$ is the aspect ratio which is one in this test problem.

Considering the properties shown in Table 5-1 and having satisfied parameters (5-10):

$$\begin{cases} \dfrac{H}{L} = 1 \\ \Pr = 1.44 \times 10^{-2} \\ \dfrac{\Pr}{0.2 + \Pr} Ra \cong 9968.44 \end{cases}$$

the resulting $\overline{Nu}$ number is shown in Table 5-2.

As it can be seen in Table 5-2, the numerical result is well matched with the experimental correlations proposed in [82] and has only 2.31% deviation.

	$\overline{Nu}$
Correlation (5-9)	2.600
Numerical Results	2.660
Percentage Error	2.31%

Table 5-2, Comparison between numerical and experimental Nusselt number

5.3- Case study 3: Solidification problems

The configuration for the first problem of this section is illustrated in Figure 5-7.

As explained previously, the main focus of this problem is on the validity of the linked VOF method and the source-based energy equation.

The basic features of the problem are explained as below:

Initially the Al-4.5% Cu in the cavity is above the eutectic temperature. At time t=0 the temperature at the surface x=0 is reduced and fixed at a temperature below the eutectic temperature so that as time proceeds a solid layer grows from this surface.

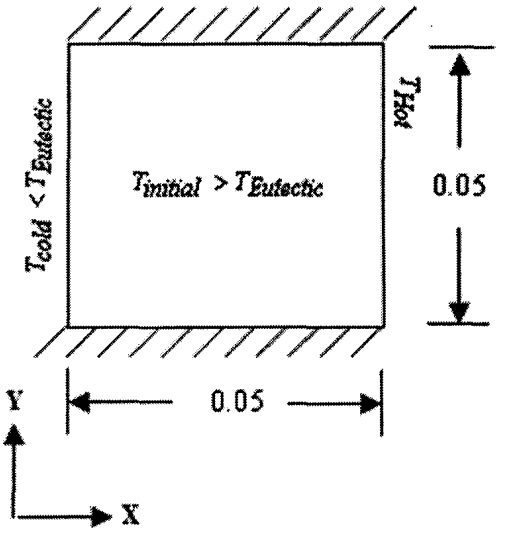

Figure 5-7, Thermal Cavity used in the Solidification test problem

Computational details

The proposed test problem is solved on a 40×40 non-uniform orthogonal grid. A fixed time step of $\Delta t = 1.0(s)$ is used in all runs. The grid size of 40×40 was obtained from a grid refinement study. Wherein, the total fraction of solid at a certain time was recorded for non-uniform grid sizes 10×10, 20×20 and 40×40. The results of this study are summarized in table 5-3.

Size	Fraction of solid at t =30
10×10	0.423
20×20	0.408
40×40	0.407

Table 5-3, Grid dependence

The permeability coefficient, κ, in the Carman-Kozeny source term as explained before depends on the morphology of the mushy region and has to be set to a value small enough to allow for significant flow in the mushy region at low local solid fraction

whereas resulting a large value of S_u to swamp out all terms in the governing equations and force any velocity predictions effectively to zero in case of having low local liquid fraction [83, 84, 85]. In the current work, the constraint set for the value of κ can be satisfied by all the values within a range between O (10) and O (10^{-3}).

Results

Figure 5-8 shows the temperature contours near the beginning of the process, t=8(s). The solidus line, i.e. the temperature isotherm $T = 760K$, shows small deformation due to convection similar in shape to the deformation predicted in isothermal phase-change cases. The liquidus front, however, shows a bulge along the lower wall. This bulge is a direct effect of the convection term in the energy equation.

Figure 5-9 shows the liquid volume fraction field and the related flow field at t= 15(s). In the liquid portion of the cavity, the velocities are in qualitative agreement with other studies of natural convection in a thermal cavity. i.e. the one presented in the previous section of this chapter.

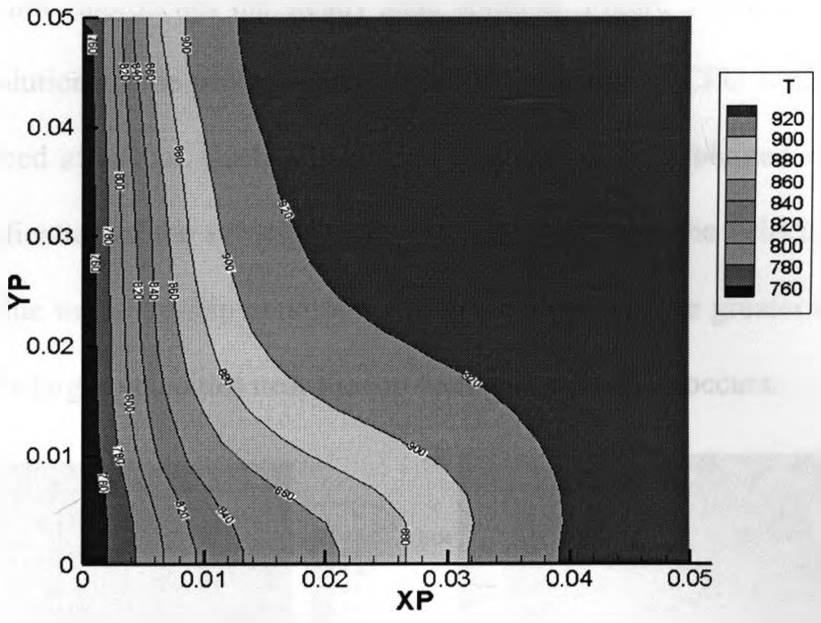

Figure 5-8, Plot shows the isotherms at t=8(s)

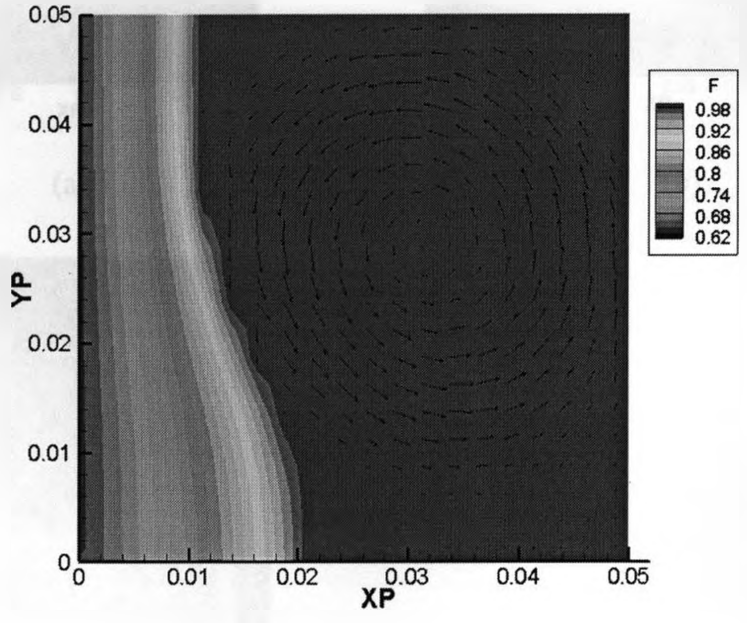

Figure 5-9, Plot shows the flow field and volume fraction field at t=15(s)

Figure 5-10 shows the development of the mushy region for different times of the process. There appears to be some re-melting of the mushy region near the top of the

cavity as the time approaches the steady state solution. Figure 5-10C is essentially the steady state solution to the problem after about 90 minutes of CPU time, as not much change is noticed after that. Such re-melting behavior can be explained in terms of the transient modification of the velocity field. At initial time steps the velocity near the top wall is small due to the no-slip condition. At later time steps, the greater momentum of the fluid creates larger velocities near the top wall and re-melting occurs.

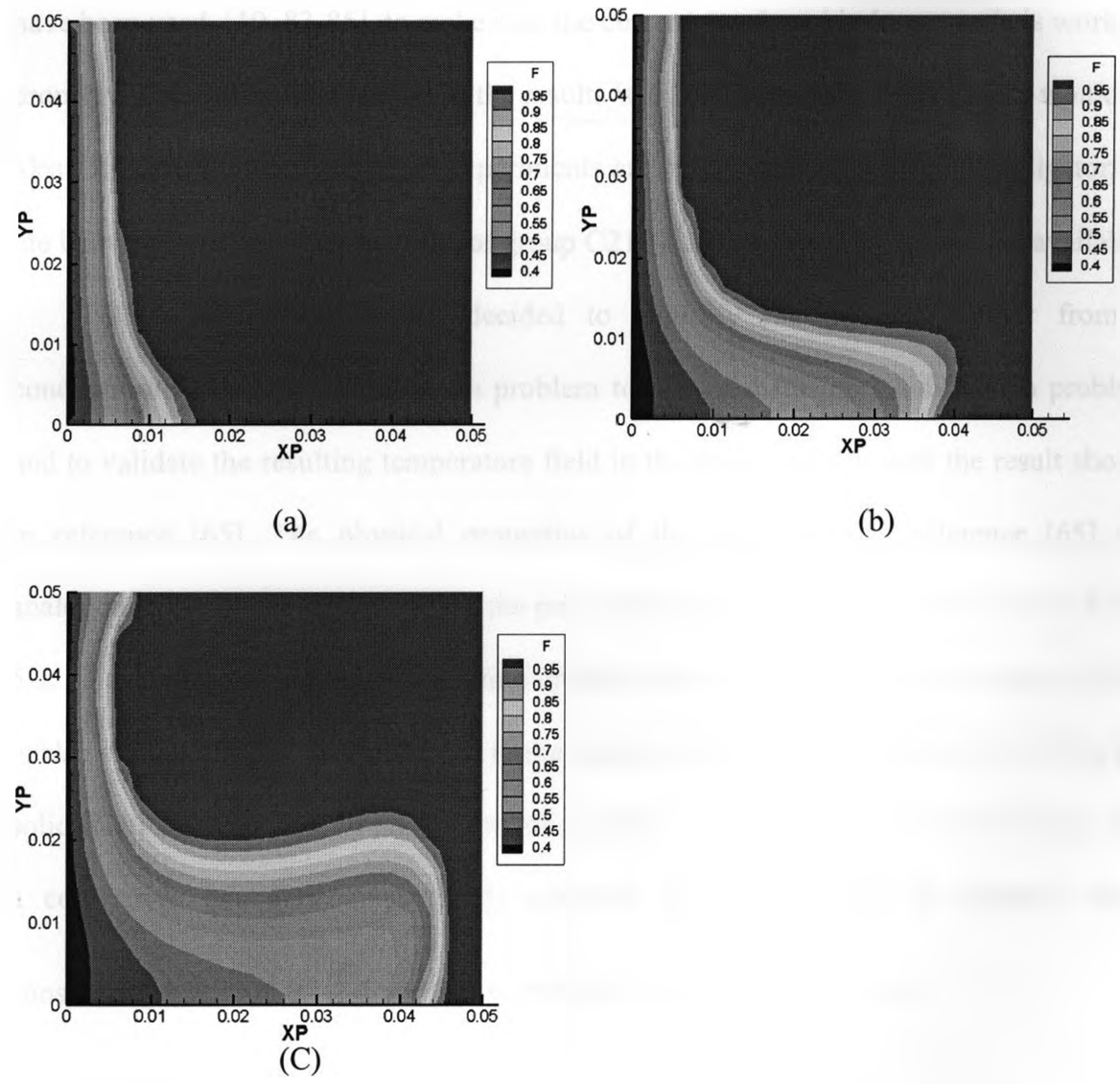

Figure 5-10, Plots show the progress of mushy region in different times, (a) t=8(s), (b) t=23(s) and (c) t=46(s)

The bulge in the mushy region is also noticeable at very early times. Since a mixed model is used for modeling the mushy region, the bulge created near the lower wall after a few time-steps shows the influence of the latent heat convection on the shape of the mushy region.

Although the results shown in Figures 5-8 to 5-10, are reasonable and are qualitatively comparable to other similar work in which the analogous transport equations have been used, [10, 83-85], to make sure the current developed in-house code is working properly, a complete comparison of the results with similar experimental results would be ideal. It is worth mentioning that experiments are being conducted by co-investigators in the University of Windsor as a part of group C210-CMG, but results are not yet available.

As a result, the author decided to simplify the problem further from a conduction/convection solidification problem to only a diffusion solidification problem and to validate the resulting temperature field in the mushy region with the result shown in reference [65]. The physical properties of the alloy used in reference [65] are analogous to what have been used in the prior problem of this section and shown in Table 5-1. The conditions and volume fraction update scheme used in the mushy region of [65] is also similar to what has been used in the current work. To follow the work of [65], the solidification is confined to one quarter of a corner region of dimension 0.1×0.1 (m), and a convective boundary condition is assumed to prevail along the exposed edges considering $h = 2000.0 \frac{W}{m^2 K}$ and $T_{\inf} = 290.0K$ as shown in Figure 5-11.

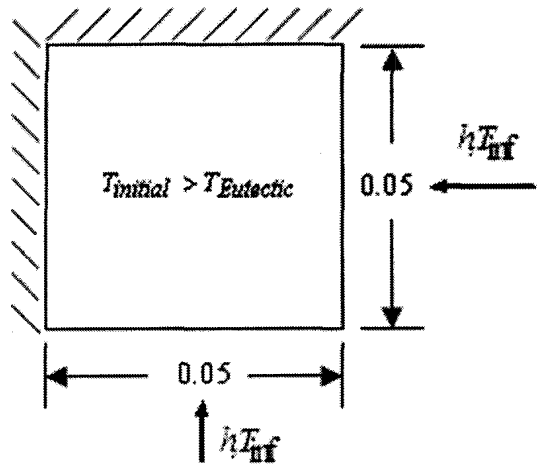

Figure 5-11, Problem specification for validation

The following tasks and changes have been made to compare the temperature fields of the current work and numerical results of reference [65]:

- A grid study has been conducted based on the value of f for a particular control volume, x=y=0.01(m), to be able to choose appropriate size of the grid. Results are plotted in Figure 5-12A. It is found that a grid size of 20×20 is appropriate to use ,
- $\Delta t = 0.5$ (s) is set instead of 1.0 (s) and
- A uniform grid is used instead of non-uniform grids.

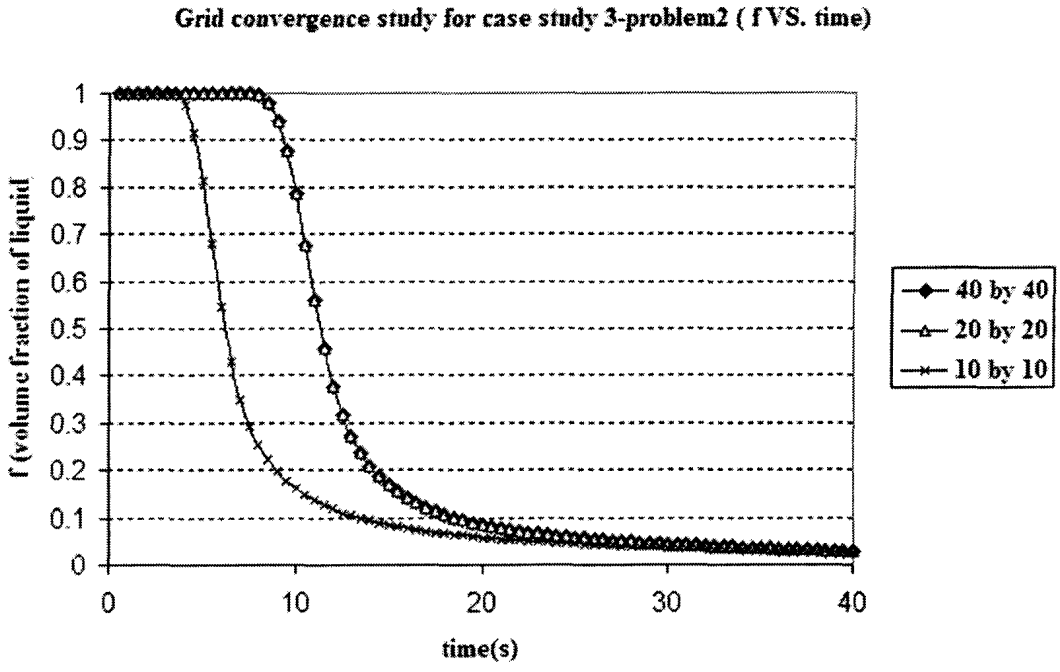

Figure 5-12A, Grid study for the second problem of the third case study

Having made these changes the result of temperature at x=y=0.01(m) is shown in Figure 5-12B.

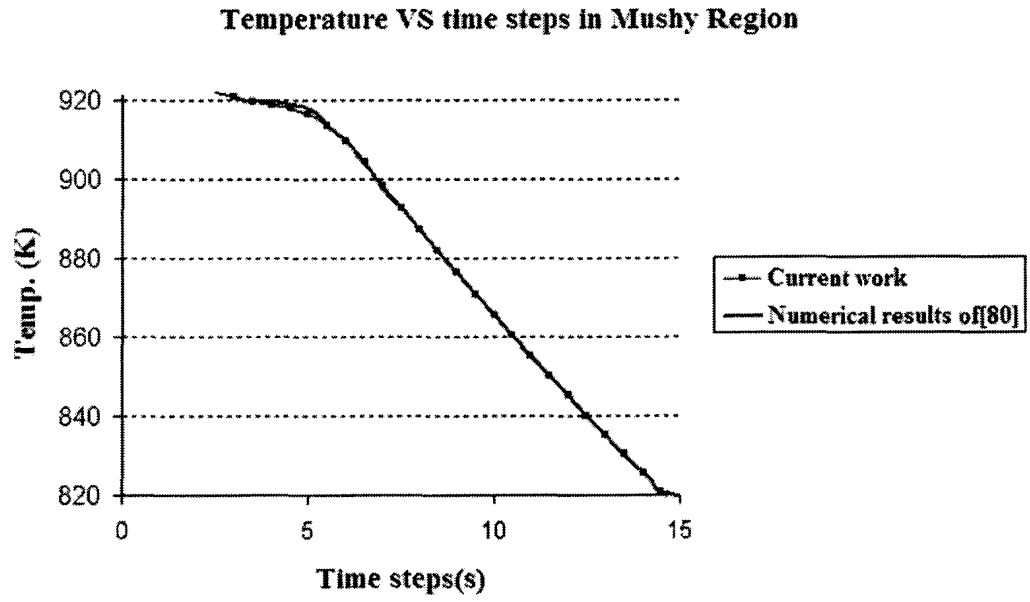

Figure 5-12B, Temperature field comparison between the current work and reference [80]

As it can be seen from the graph shown in Figure 5-12B, the temperature field in the mushy region is well matched with the result obtained in previous work of Swaminathan and Voller [65] and this confirms that the code developed in the current work is accurate.

The author used the modified code of the second problem of this section to test a case similar to the first problem of the current case study.

Properties of Al-6%Mg	Units	Value
Specific heat	$C_s\ [\frac{J}{kgK}]$	1097
	$C_s\ [\frac{J}{kgK}]$	992
Thermal conductivity	$K_s\ [\frac{W}{mK}]$	130
	$K_l\ [\frac{W}{mK}]$	92.6
Density	$\rho\ [\frac{\text{kg}}{\text{m}^3}]$	2315
Viscosity	μ [Kg/m s]	1.17×10^{-4}
Latent heat of fusion	L [J/kg]	3.96×10^{5}
Eutectic temperature	T_E [K]	723
Melting temperature	T_m [K]	933.5
Liquidus temperature	T_l [K]	919
Thermal expansion	$\beta[1/K]$	2.5×10^{-6}
Partition coefficient	K_P	0.472

Table 5-4, Aluminum-Magnesium Data Properties of binary Al-6% Mg Alloy [81]

The only differences are: a different binary alloy, Al-6%Mg, with properties shown in Table 5-4, is used in the cavity shown in figure 5-6, and the convection terms are dropped from the energy transport equation.

Figure 5-13 shows the result of this test problem. Because there is not any flow to drive the temperature field or value of f similar to the first problem of the third case study, it can be seen that the bulge created in Figure 5-10 due to the effect of convection terms does not exist any more in this case which is also another expected result.

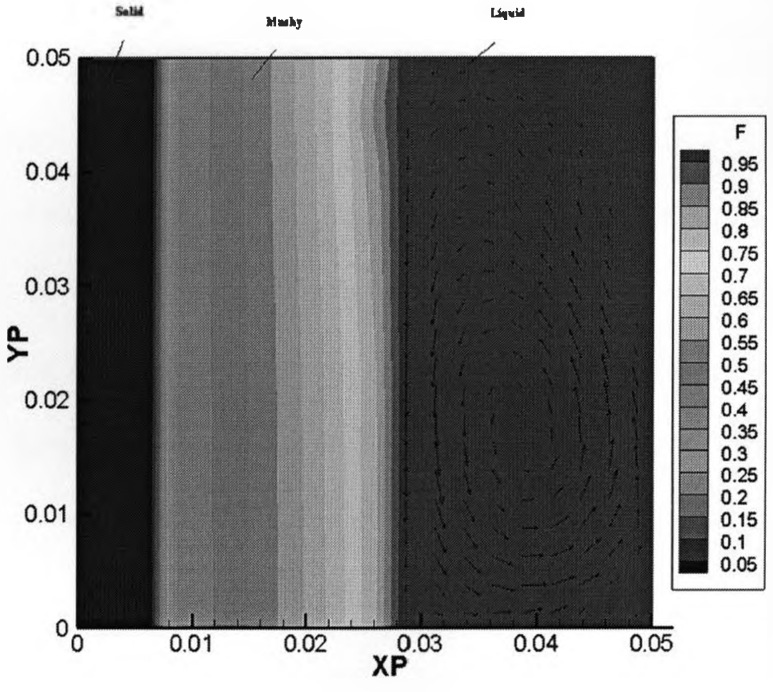

Figure 5-13, Temperature contours and velocity vectors for a simple solidification problem wherein the domain is cooled from the left and heated from the right at t=48(s).

In this chapter the formulation of the solidification model and its implementation have been validated.

5.4- Summary

In summary the formulation used in the present study, the interface-capturing method selected, and the changes made to enable smooth pressure transition and interface movement are seen to give very good results. This code forms the basis for future work to be conducted on solidification problems with more complex geometry and filling processes. Author will summarize the current work and suggest a few future contributions in the next chapter.

Chapter 6

6-Summary and Future contributions

6.1-Summary

The principle aim of this work was to develop a formulation that provides an excellent platform for simulation of solidification processes and filling processes. This motivated the development of a fixed grid approach along with retaining the basic form of the hydro-mechanical equations. The phenomena associated with the solidification were modeled by careful consideration of the essential physics. The driving source terms are Carman-Kozeny source terms in the momentum equations and the latent heat source terms in the energy equation. The choice of Carman-Kozeny source terms originated from the effect of the porosity of the mushy region on the flow field. The latent heat source term is a function of the liquid/solid fraction which is a function of temperature. In this work, the changes in volume fractions follow the Scheil equation. An in-house VOF based code was developed and added to the solution algorithm to be able to track the interface between constituents, attain the best normal approximation to the interface and create an excellent platform for future simulation of a filling process in which tracking the liquid-gas-solid interface in the domain is necessary. The overall formulation is simple to implement and upgrade to 3D, and can be compatible with most numerical packages available.

6.2- Present Contribution

- In this work, the author presents a formulation that creates not only a platform for the modeling of mold filling and solidification of binary alloys, but also provides

an excellent platform to capture the interface between constituents in a die-casting process including solidification and mold filling process.

- A volume-of-fluid (VOF) based method to capture the interface between solid and liquid in a solidification process on a fixed non-uniform grid, developed for implementation in a co-located finite volume framework, was presented.
- Differing from other works, to update the volume fraction of fluid in the field, a link between source-based type of energy equation and VOF algorithm is described and implemented.
- A new approximation to the pressure gradient in the momentum equation is presented to remove all "Spurious Currents" [1] in the vicinity of the interface.

6.2- Future contributions

There is a need for further studies to be made. In particular:

- An investigation into different approaches and models for flow in the mushy zone; the questions that researchers should be able to answer are; what is an appropriate form for the morphology- porosity relationship in the mushy zone? Is the Carman-Kozeny law appropriate to use? An investigation of this type could have a significant effect on applications of the current formulation to metal systems, where the flow in the mushy region is important.
- Some experimental studies are required. The work presented lacks experimental validation for solidification cases. After obtaining experimental results by co-investigators at University of Windsor, the numerical results must be compared with these experimental results prior to further code modifications.

- Researchers should consider the effects of shrinkage and macro-segregation in future modeling of solidification problems.
- Research needs to be conducted to find the most appropriate form of source terms in the transport equations. Indeed as it stands its framework makes it an ideal platform for the current understanding of the solidification process.

References

[1] S. Popinet and S. Zaleski, A front-tracking algorithm for the accurate representation of surface tension, *International Journal for Numerical Methods in Fluids,* Vol. 30, PP.775–793, (1999)

[2] A. K. Dahle, S. Sannes, D.H. St. John and H. Westengen, Formation of defect bands in high pressure die cast magnesium alloys, *Journal of Light Metal,* Vol. 1, PP. 99-103, (2001)

[3] K.U. Kainer, *Magnesium Alloys and Technologies,* WILEY-VCH GmBH &Co. KGaA, Weinheim, (2003).

[4] G. Schindelbacher and R. Rosch, Mechanical properties of magnesium die- casting alloys at elevated temperature and microstructure in dependence of wall thickness, *Magnesium Alloys and their application*, German, PP. 247-252, (1998)

[5] J. Crank *Free and Moving boundary problems*, Oxford: Clarendon, (1984)

[6] H. Hu, S. A. Argyropoulos, Mathematical modeling of solidification and melting: A review, *Modeling and Simulation in Material Science and Engineering*, Vol. 4, PP. 371-396, (1996)

[7] V. R. Voller Numerical Methods for phase-change problems, in: edited by W.J. Minkowycz, E.M. Sparrow, J. Y. Murthy; with editorial assistance by John P. Abraham, *Handbook of numerical Heat transfer*, 2nd ed. Hoboken, N.J : J. Wiley, PP. 593-622, (2006)

[8] M. McDonald, Engineering Material note book, British Columbia Institute of Technology, Vancouver, Canada, (2006)

[9] C. Beckermann, R. Viskanta, Mathematical modeling of transport phenomena during alloy solidification, *Applied Mechanics Review,* Vol. 46, No.1, PP. 1-27, (1993)

[10] V.R. Voller, A.D. Brent and C. Prakash, Modeling the mushy region in a binary alloy, *Applied Mathematical Modeling*, Vol.14, PP. 320-326, (1990)

[11] V.R. Voller, A.D. Brent and C. Prakash, The modeling of heat, mass and solute transport in solidification system, *International Journal of Heat and Mass Transfer*, Vol. 32, PP. 1719-1731, (1989)

[12] E. Mcbride, J. C. Heinrich and D. R. Poirier, Numerical simulation of incompressible flow driven by density variations during phase change, *International Journal for Numerical Methods in Fluids*, Vol. 31, PP 787-800 (1999)

[13] J. F. Thompson, Numerical solution of flow problem using Body-Fitted-coordinate systems, in W. Kollmann (ed.), *Computational Fluid Dynamics*, PP.1-98, Hemisphere, Washington, D.C., (1980)

[14] J. F. Thompson, Grid Generation, in W. J. Minkowycz, E. M. Sparrow, G. E. Schneider, and R. H. Pletcher (eds.), *Handbook of Numerical Heat Transfer*, PP.905-948, Wiley-Interscience, New Yourk, (1988)

[15] M. Lacroix and V. R. Voller, Finite difference solution of solidification phase change problems: Transformed versus fixed grids, *Numerical Heat Transfer Part B*, Vol.17, PP. 25-41, (1990)

[16] D. E. Fyfe, E.S. Oran and M.J. Fritts, Surface tension and viscosity with Lagrangian hydrodynamics on a triangular mesh, *Journal of Computational Physics*, Vol.76, PP. 349–384 (1988).

[17] J. Glimm, O. McBryan, R. Menikoff and D.H. Sharp, Front tracking applied to Rayleigh–Taylor instability, *SIAM Journal on Scientific and Statistical Computing* Vol. 7, PP. 230–251 (1987).

[18] W. D. Bennon and F. P. Incropra, A Continuum model for momentum, heat and Species transport in Binary Solid-Liquid phase change systems I: Model formulation, *International Journal of Heat and Mass Transfer,* Vol. 30, PP. 2077-2089, (1987)

[19] W. D. Bennon and F. P. Incropra, A Continuum model for momentum, heat and Species transport in Binary Solid-Liquid phase change systems II: Application to Solidification in a rectangular Cavity, *International Journal of Heat and Mass Transfer,* Vol. 30, PP. 2171-2187, (1987)

[20] J. M. Hyman, Numerical methods for tracking interfaces, *Physica D* Vol.12, PP.396-407, (1984)

[21] S. H. Unverdi and G. Tryggvason, A front-tracking method for viscous, incompressible, multi-fluid flows, *Journal of Computational. Physics* Vol.100, PP. 25 (1992)

[22] M. Sussmann and P. Smereka, Axisymmetric free boundary problems, *Journal of Fluid Mechanics,* Vol. 341, PP.269, (1997)

[23] W. J. Rider and D. B. Kothe, Reconstructing volume tracking, *Journal of Computational Physics* Vol. 141, PP.112 (1998)

[24] M. Rudman, Volume tracking methods for interfacial flow calculations. *International Journal for Numerical Methods in Fluids,* Vol.24, PP.671, (1997)

[25] M. Rudman, Volume tracking methods for incompressible multi-fluid flows with large density variations, *International Journal for Numerical Methods in Fluids,* Vol. 28, PP.357–378, (1998)

[26] J. H. Ferziger and M. Peric, Computational Methods for fluid Dynamics, Springer, Berlin, (2002)

[27] J. Glimm, J. Grove, B. Lindquist, O. McBryan, and G. Tryggvason, The bifurcation of tracked scalar waves, *SIAM Journal on Scientific and Statistical Computing* Vol. 9, PP. 61 (1988).

[28] B. J. Daly, A technique for including surface tension effects in hydrodynamic calculations, *Journal of Computational Phyics,* Vol. 4, PP.97 (1969).

[29] A. Takizawa, S. Koshizuka, S. Kondo and S. Generalization of physical component boundary fitted co-ordinate, (PCBFC) method for the analysis of free-surface flow. *International Journal of Numerical Methods in Fluids*, Vol. 15, p. 1213-1237, (1992)

[30] B. D. Nichols, and C.W Hirt, Calculating three-dimensional free surface flows in the vicinity of submerged and exposed structures. *Journal of Computational Physics*, Vol. 12, PP. 234-246, (1973)

[31] A. Dervieux, and F. Thomasset, A finite element method for the simulation of a Rayleigh-Taylor instability. *IRIA-LABORIA report,* F-78150 Le Chesnay, (1979)

[32] J. Glimm, J. Grove, B. Lindquist, O. A. McBryan, and G. Tryggvason, The bifurcation of tracked scalar waves, *SIAM Journal on Scientific and Statistical Computing*, Vol. 9(1), PP. 61-79, (1998)

[33] J. Glimm, O. McBryan, R. Menikoff, and D.H. Sharp, Front tracking applied to Rayleigh-Taylor instability. *SIAM Journal on Scientific and Statistical Computing*, Vol. 7(1), PP. 230-251, (1986)

[34] E. G. Puckett, A. S. Almgren, J. B. Bell, D. L Marcus, W. J.Rider, High Order Projection Method for Tracking Fluid Interface in Variable Density Incompressible Flows, *Journal of Computational Physics,* Vol. 130. PP. 269-282, (1997)

[35] F. H. Harlow and J. E. Welsh, Numerical calculation of time dependent viscous incompressible flow with free surface, *Physics of Fluids*, Vol. 8, PP. 2182-2189 (1965)

[36] C. W. Hirt and B. D. Nichols, Volume of Fluid Method for the Dynamics of Free Boundaries, *Journal of Computational Physics*, Vol. 39, PP. 201-225 (1981)

[37] S. Osher, and J. A. Sethian, Fronts propagating with curvature-dependant speed: algorithms based on Hamilton-Jacobi formulations. *Journal of Computational Physics*, Vol. 79, PP. 12-49, (1988)

[38] M. Sussman, P. Smereka, and S. Osher, A level set approach for computing solutions to incompressible phase flow, *Journal of Computational Physics*, Vol. 114, PP. 146-159, (1994)

[39] B.D. Nichols, C.W. Hirt, R.S. Hotchkiss, SOLA-VOF: a solution algorithm for transient fluid flow with multiple free boundaries, *Technical Report LA-8355*, Los Alamos National Laboratory, (1980)

[40] M.D. Torrey, L.D. Cloutman, R.C. Mjolsness, C.W. Hirt, NASA-VOF2D: a computer program for incompressible flows with free surfaces, *Technical Report LA-10612-MS*, Los Alamos National Laboratory, (1985)

[41] D.B. Kothe and R.C. Mjolsness, RIPPLE: a new model for incompressible flows with free surfaces. *American Institute of Aeronautics and Astronautics Journal,* 30 11, PP. 2694–2700, (1992)

[42] D.B. Kothe, R.C. Mjolsness, M.D. Torrey, RIPPLE: a computer program for incompressible flows with free surfaces, *Technical Report LA-12007*-MS, Los Alamos National Laboratory, (1991)

[43] W.F. Noh and P.R. Woodward, SLIC (Simple Line Interface Calculation). In: A.I. van der Vooren and P.J. Zandbergen, Editors, *Lecture Notes in Physics* Vol. **59**, Springer-Verlag, New York, pp. 330–340, (1976)

[44] L. Margolin, J.M. Reisner and P.K. Smolarkiewicz, Application of the volume-of-fluid method to the advection-condensation problem. *Monthly. Weather Review* Vol.125, PP. 2265–2273, (1997)

[45] R. DeBar, A method in 2-D Eulerian hydrodynamics, *Technical Report UCID-19683*, Lawrence Livermore National Laboratory, (1974)

[46] L.F. Henderson, P. Colella and E.G. Puckett, On the refraction of shock waves at a slow-fast gas interface, *Journal of Fluid Mechanics* 224, pp. 1–27, (1991)

[47] E.G. Puckett, A numerical study of shock wave refraction at a CO_2/CH_4 interface. In: J. Glimm and A.J. Majda, Editors, *Multidimensional Hyperbolic Problems and Computations IMA Volumes in Mathematics and Its Applications* Vol. 29, Springer-Verlag, pp. 261–280, (1991)

[48] J. E. Pilliod and E. G. Puckett , Second-Order accurate volume-of-fluid algorithms for tracking material interfaces, *Journal of Computational Physics* Vol. 199 PP. 465-502, (2004)

[49] A. J. Chorin, Flame advection and propagation algorithms. *Journal of Computational Physics*, Vol. 35, PP. 1-11, (1980)

[50] D.L. Youngs, Time-dependent multi-material flow with large fluid distortion. In: Morton, K.W. and Baines M.J. (eds.) *Numerical Methods for fluid dynamics*. London: Academic Press. PP. 273-285, (1982)

[51] N.Ashgriz and J.Y. Poo, FLAIR : Flux line-segment model for advection and interface reconstruction, *Journal of Computational Physics,* Vol. 93, PP. 449-468, (1991)

[52] M. Sussman and E.G. Puckett, A coupled level set and volume of fluid method for computing 3d and axisymmetric incompressible two-phase flows. *Journal of Computational Physics* Vol.162, PP. 301–337, (2000)

[53] E. Shirani, N. Ashgriz, J. Mostaghimi, Interface pressure calculation based on conservation of momentum for front capturing methods, *Journal of Computational Physics* Vol. 203, PP. 154-175, (2005)

[54] V. R. Voller, C. R. Swaminathan and B. G. Thomas, Fixed Grid Techniques for phase change problems: A Review, *International Journal for Numerical Methods in Engineering,* Vol. 30, PP. 875-898 (1990)

[55] M. Zerroukat, and C. R. Chatwin Computational moving boundary problem Research studies Press, England, (1994)

[56] J. S. Hsiao, An efficient algorithm for finite difference analyses of heat transfer with melting and solidification, *Numerical Heat Transfer,* Vol.8, PP. 653-666, (1985)

[57] D. Celentano, M. Cruchaga, N. Moraga, and J. Fuented, Modeling natural convection with solidification in mold cavities, *Numerical Heat Transfer, Part A,* Vol. 39, PP. 631-654, (2001)

[58] M. Saludean, and Z. Abdullah, on the numerical modeling of heat transfer during solidification process, *International Journal for numerical Methods in Engineering,* Vol. 25, PP. 445-473, (1988)

[59] D. Poirier and M. Salcudean, On Numerical methods used in mathematical modeling of phase change in liquid metals, *Master of Applied Science Thesis, University of Ottawa,* (1986)

[60] N. Shamsunder, and E. M. Sparrow, Analysis of Multidimensional conduction phase change via the enthalpy model, *Journal of Heat Transfer,* PP.333-340 (1975)

[61] A. D. Solomon, V. Alexiades, D. G. Geist and J. Kerper, An enthalpy method for the solidification of a super-cooled liquid, *ORNL-6218,* (1986)

[62] V. Voller, Implicit finite difference solution of the enthalpy formulation of Stefan problems, *IMA Journal of Numerical Analysis,* Vol. 5, PP. 201-214 (1985)

[63] V. R. Voller, Interpolation of the enthalpy in descretized multidimensional region undergoing a phase change, *International Communication. Heat Mass Transfer,* Vol. 10, PP. 323-328 (1983)

[64] V. R. Voller and Cross, An explicit numerical method to track a moving phase change front, *International Journal of Heat and Mass Transfer,* Vol.26, PP. 5454-556 (1980)

[65] C. R. Swaminathan and V. R. Voller, A general enthalpy method for modeling solidification Process, *Metallurgical Transactions B,* Vol. 23B, PP. 651,664, (1992)

[66] V. R. Voller and C. R. Swaminathan, Earl Source-Based Method for solidification phase change, *Numerical Heat Transfer, Part B: Fundamentals,* Vol. 19, PP. 175-189, (1991)

[67] D. R. Poirier, Permeability for flow of interdendritic liquid in columnar-dendritic alloys, *Metallurgical and Materials Transactions B* , Vol. 18b, PP.245-256, (1987)

[68] J. Ni and C. Beckermann, A volume averaged two phase model for transport phenomena during solidification, *Metallurgical Transactions B*, Vol. 22B, PP. 349-361, (1991)

[69] M. Rappaz, Modeling of microstructure formation in solidification process, *International Material Review,* Vol. 34, PP. 93-123 (1989)

[70] D. M. Stefanescu and C. S. Kanetkar, Modelling microstructural evolution of eutectic cast iron and of the gray/white transition, *Material science and Engineering,,* Vol. 95, PP. 139-144 (1987)

[71] C. A. Siqueira, N. Cheung and A. Garcia, The Columnar to equiaxed Transition during solidification of Sn-Pb alloys, *Journal of Alloys and Compounds,* Vol. 351, PP. 126-134 (2003)

[72] S. V. Patankar, Numerical Heat Transfer and Fluid flow. Hemisphere Publishing: Washington. DC (1980)

[73] C. M. Rhie and W.L. Chow, Numerical Study of the Turbulent flow Past an Airfoil with Trailing Edge Separation, *American Institute of Aeronautics and Astronautics Journal,* vol. 21, pp. 1525-1532, (1983)

[74] V. R. Voller, development and application of a heat balance integral method for analysis of metalorgical solidification, *Applied Mathematical Modeling*, Vol.13, PP. 702-709, (1989)

[75] M. Cross, S. Johnson and P. Chow, Mapping enthalpy-based solidification algorithm onto vector and parallel architecture, *Applied Mathematical Modeling*, Vol. 13, PP.702-709, (1989)

[76] DB. Kothe Perspective on Eulerian finite volume methods for incompressible interfacial flows, In *Free Surface Flows*, Kuhlmann HC, Rath HJ (eds). Springer: Berlin, PP. 267–331 (1997).

[77] D. K. Agarwal, S. W. J. Welch, G. Biswas, F. Durst, Planar Simulatoin of Bubble Growth in film Boiling in Near-Critical Water Using a Variant of VOF method, *Journal of heat transfer,* Vol. 126, PP. 329-338, (2004)

[78] S. W. J. Welch and T. Rachidi, Numerical Computation of film Boiling Including Conjucate Heat Transfer, *Numerical Heat Transfer, PartB,* Vol. 42, PP. 35-53, (2002)

[79] W. H. Press, S. A. Teukolsky, W. T. Vetterling, And B. P. Flannery, *Numerical Recipes in Fortran,* Cambridge,(1986)

[80] S. S. Clift, E. F. D'Azevedo, P. A. Forsyth and J. R. Knightly, WATSIT-1 and WATSIT-B, Waterloo Sparse, Iterative Matrix solvers user's guide with developer notes for version 2.0.0 (1997)

[81] C. J. Vreeman and F. P. Incropera, The effect of free-floating dendrites and convection on Macro-segregation in direct chill cast aluminum alloys Part II: predictions for Al-Cu and Al-Mg alloys, *International Journal of Heat and Mass Transfer*, Vol. 43, PP.687-704, (2000)

[82] F. P. Incropera, D. P. Dewitt, T. L. Bergman and A. S. Lavine, Fundamental of heat and mass transfer, 6th edition, John Wiley & Sons, NJ, (2006)

[83] V.R Voller and C. Prakash, A fixed grid numerical modeling methodology for convection-diffusion mushy region phase-change problems, *International Journal of Mass Transfer*, Vol. 30, PP.1709-1719, (1987)

[84] V. R. Voller, M. Cross and N.C. Markatos, An enthalpy method for convection/diffusion phase change, *International Journal for Numerical Methods in Engineering*, Vol. 24, PP. 271-284, (1987)

[85] G. Arampatis, D. Assimacopoulos, Numerical modeling of convection-diffusion phase change problems, *Journal of Computational Mechanics*, Vol.21, PP. 409-415 (1998)

Appendix A

The volume of fluid advection algorithm used in general VOF methods

The case is considered as a problem of advancing an interface in a known divergence-free velocity field $\vec{u} = (u, v)$ solved in the x- y- momentum equations. With each grid cell a volume fraction f is associated that indicates the portion of the cell that is occupied by fluid; i.e.,

$$f_{i,j}\Delta x\Delta y = \text{Volume of fluid in cell } (i, j) \tag{1}$$

The number $f_{i,j}$ is called the volume fraction (of fluid) in the (i, j) th cell. It is obvious that $0 \le f_{i,j} \le 1$. The volume fraction of the other material is $(1 - f_{i,j})$. The discretization of the characteristic function associated with the fluid considering the fact that [49],

$$f_{i,j}\Delta x\Delta y \approx \iint_{i,jcell} f(x,y)dxdy \text{ , is } f(x,y) = \begin{cases} 1 & \text{if there is fluid at point (x, y)} \\ 0 & \text{if there is solid at point(x, y)} \end{cases}.$$

The characteristic function, also it is called color function, is passively advected with the flow because the fluid type does not change along particle path. It can be concluded f satisfies the advection equation,

$$f_t + uf_x + vf_y = 0 \tag{2}$$

$\vec{u}$ satisfies equation (3) if the flow under consideration is incompressible.

$$\nabla.\vec{u} = 0 \quad \text{or} \quad u_x + v_y = 0 \tag{3}$$

By multiplying the recent equation by f and adding it to the advection equation, a conservation law for the characteristic function can be obtained [48];

$$f_t + (uf)_x + (vf)_y = 0 \tag{4}$$

This means that material volumes remain constant on streamlines. In other words, the incompressibility of the flow allows that the conservation of volume to be equal to the conservation of mass [48].

Integration of material volume evolution equation requires the evaluation of material volume fluxes at each cell face. These fluxes represent the volume of constituent passing through a face during the current time step. The sum of all material or constituent volume fluxes must be equal to the total volume flux, which for example for east face of a selected cell is [49]

$$\sum dV_{i+1/2,j} = V_{i+1/2,j} = (\Delta t.\vec{u}_{i+1/2,j}) \times \vec{A}_{i+1/2,j} \tag{5}$$

where, $\vec{u}$ is the velocity field and $\vec{A}_{i+1/2,j}$ is the edge area vector and in a 2D case is the width of a computational cell, Δy.

To compute $V_{i+1/2,j}$, two main steps are required:

- Construction of the polygon bounding the volume swept by the velocity field during time step
- Finding the amount of fluid or material inside this polygon.

The first step can be easily accomplished by the reconstruction algorithm explained in details in Chapter 4; however, the second step needs some definitions.

Consider a mixed cell as shown in Figure A. Based on the face velocity at the east face of the cell, a portion $(\Delta t.\vec{u}_{i+1/2,j})\times\Delta y$ of the control volume can be calculated, whose fluid will be fluxed into the cell at the east of this mixed cell. The flux can be calculated as bellow [23];

$$f_e\vec{u}_e = \frac{V_{i+1/2,j}\times\vec{u}_{i+1/2,j}}{V_{rectangle}} = \frac{V_{i+1/2,j}\times\vec{u}_{i+1/2,j}}{\vec{u}_{i+1/2,j}\times\Delta t\times\Delta y} = \frac{V_{i+1/2,j}}{\Delta t\times\Delta y} \qquad (6)$$

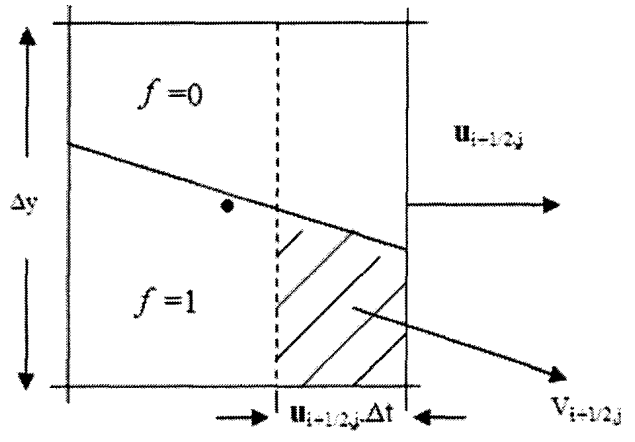

Figure A, Schematic of a computational cell to show fluid flux calculation

There are two types of advection algorithms which are used more often in recent works to update f. One is the "Operator Split algorithm" and the other one is "Unsplit advection algorithm" [48]. The simplest of two advection algorithm is the former algorithm which is chosen to be explained bellow;

Using the usual discretization scheme introduced in Patankar [72], Integrating $\frac{\partial f}{\partial t} + \nabla.(f\vec{u}) = 0$ over a cell associated with node P and over the time step Δt, the discretized advection equation for f is given by:

$$\begin{aligned}
&\frac{(f^{n+1} - f^n)\Delta x\Delta y}{\Delta t} + \sum_i (fuA)_i = 0 \\
&\frac{(f^{n+1} - f^n)\Delta x\Delta y}{\Delta t} + (f_e\vec{u}_e A_e - f_w\vec{u}_w A_w) + (f_n\vec{u}_n A_n - f_s\vec{u}_s A_s) = 0 \qquad (7a) \\
&\frac{(f^{n+1} - f^n)\Delta x\Delta y}{\Delta t} + (f_e\vec{u}_e - f_w\vec{u}_w)\Delta y + (f_n\vec{u}_n - f_s\vec{u}_s)\Delta x = 0
\end{aligned}$$

$$f^{n+1} = f^n + (f_w\vec{u}_w - f_e\vec{u}_e)\frac{\Delta t}{\Delta x} + (f_s\vec{u}_s - f_n\vec{u}_n)\frac{\Delta t}{\Delta y} = 0 \qquad (7b)$$

However, for an operator split time integration scheme, the original equation, (4) , must be integrated twice in two dimension (thrice in three dimension), one integration per each sweep and one sweep per dimension. So, equation (7) looks like [48, 51]

$$\tilde{f}_{i,j} = f^n_{i,j} + (f_w\vec{u}_w - f_e\vec{u}_e)\frac{\Delta t}{\Delta x} \qquad (8a)$$

$$f^{n+1}_{i,j} = \tilde{f}_{i,j} + (f_s\vec{u}_s - f_n\vec{u}_n)\frac{\Delta t}{\Delta y} \qquad (8b)$$

where $\tilde{f}$ is the intermediate volume fraction which is obtained by using equation (6) in (8). The vertical fluxes can be obtained with the same geometric construction that explained to attain equation (6) . Rider and Kothe [23] showed that using a modified form of original conservation form of f equation can overcome the overshoot and undershoot

of f during calculation. They suggested a new form of operator split scheme with "divergence correction". Technically in this scheme the solution of following equation

$$f_t + (fu)_x = fu_x \tag{9a}$$

$$f_t + (fv)_y = fv_y \tag{9b}$$

is approximated instead of

$$f_t + (fu)_x = 0 \tag{10a}$$

$$f_t + (fv)_y = 0 \tag{10b}$$

and in order to maintain conservation of f, on the right hand side of equation (9) and (10), f is necessary to be descretized implicitly and explicitly, respectively. The final form of descretized equation using modified operator split would be;

$$\tilde{f}_{i,j} = \frac{\left\{ f_{i,j}^{n} + (f_w \vec{u}_w - f_e \vec{u}_e)\frac{\Delta t}{\Delta x} \right\}}{\left[1 - (\vec{u}_e - \vec{u}_w)\frac{\Delta t}{\Delta x} \right]} \tag{11a}$$

$$f_{i,j}^{n+1} = \tilde{f}_{i,j} \times \left\{ 1 + (\vec{u}_n - \vec{u}_s)\frac{\Delta t}{\Delta y} \right\} + (f_s \vec{u}_s - f_n \vec{u}_n)\frac{\Delta t}{\Delta y} \tag{11b}$$

To obtain second-order accuracy it is necessary to alternate the sweep directions at each time step, taking care that in the first sweep f is differenced implicitly and in the second sweep f is differenced explicitly. In other words, in the first time step equations (11) are applied and in the time step after, the following equations [23];

$$\widetilde{f}_{i,j} = \frac{\left\{ f_{i,j}^{n} + (f_s \vec{u}_s - f_n \vec{u}_n) \frac{\Delta t}{\Delta y} \right\}}{\left[1 - (\vec{u}_n - \vec{u}_s) \frac{\Delta t}{\Delta y} \right]} \tag{12a}$$

$$f_{i,j}^{n+1} = \widetilde{f}_{i,j} \times \left\{ 1 + (\vec{u}_e - \vec{u}_w) \frac{\Delta t}{\Delta x} \right\} + (f_w \vec{u}_w - f_e \vec{u}_e) \frac{\Delta t}{\Delta x} \tag{12b}$$